



UNIVERSIDADE FEDERAL DO CEARÁ
INSTITUTO UNIVERSIDADE VIRTUAL
CURSO DE GRADUAÇÃO EM SISTEMAS E MÍDIAS DIGITAIS

NATANAEL FRANCISCO FERREIRA DE CARVALHO

**EMPREGO DE TECNOLOGIAS DE GAME PARA A RENDERIZAÇÃO EM TEMPO
REAL DE PROJETOS ARQUITETÔNICOS**

FORTALEZA

2022

NATANAEL FRANCISCO FERREIRA DE CARVALHO

EMPREGO DE TECNOLOGIAS DE GAME PARA A RENDERIZAÇÃO EM TEMPO REAL
DE PROJETOS ARQUITETÔNICOS

Trabalho de Conclusão de Curso apresentado ao Curso de Graduação em Sistemas e Mídias Digitais do Instituto Universidade Virtual da Universidade Federal do Ceará, como requisito parcial à obtenção do grau de bacharel em Sistemas e Mídias Digitais.

Orientador: Prof. Dr. Adriano Anuncia-ção
Oliveira

FORTALEZA

2022

Dados Internacionais de Catalogação na Publicação
Universidade Federal do Ceará
Sistema de Bibliotecas
Gerada automaticamente pelo módulo Catalog, mediante os dados fornecidos pelo(a) autor(a)

C326e Carvalho, Natanael Francisco Ferreira de.
Emprego de tecnologias de game para a renderização em tempo real de projetos arquitetônicos / Natanael Francisco Ferreira de Carvalho. – 2022.
57 f. : il. color.

Trabalho de Conclusão de Curso (graduação) – Universidade Federal do Ceará, Instituto UFC Virtual, Curso de Administração, Fortaleza, 2022.
Orientação: Prof. Dr. Adriano Anunciação Oliveira.

1. Computação gráfica. 2. Renderização. 3. Arquitetura. I. Título.

CDD

NATANAEL FRANCISCO FERREIRA DE CARVALHO

EMPREGO DE TECNOLOGIAS DE GAME PARA A RENDERIZAÇÃO EM TEMPO REAL
DE PROJETOS ARQUITETÔNICOS

Trabalho de Conclusão de Curso apresentado ao Curso de Graduação em Sistemas e Mídias Digitais do Instituto Universidade Virtual da Universidade Federal do Ceará, como requisito parcial à obtenção do grau de bacharel em Sistemas e Mídias Digitais.

Aprovada em:

BANCA EXAMINADORA

Prof. Dr. Adriano Anunciação Oliveira (Orientador)
Universidade Federal do Ceará (UFC)

Prof. Me. Neil Armstrong Rezende
Universidade Federal do Ceará (UFC)

Prof. Mauro de Castro Silva (Membro Externo)
Universidade de Salvador (UNIFACS))

À minha família, pelo o apoio constante que conferem a mim. Aos meus pais pela dedicação e investimento. Aos meus irmãos pela companhia de vida.

AGRADECIMENTOS

Qualquer palavra seria insuficiente para expressar minha gratidão àqueles que comigo estiveram nessa caminhada. Todavia, a fim de dedicar esse momento da minha vida para as pessoas especiais que me apoiam, inicio os meus agradecimentos.

Inicialmente, dedico esse passo importante da minha vida àquele que comigo esteve do início ao fim, que me dá abundantemente força, discernimento e mansidão nos dias de luta, ao meu Deus. Sem a ajuda divina nada disso se concretizaria.

Aos meus pais, Laudecy Ferreira e Aldísio Carvalho, aos quais pude compartilhar as dificuldades da vida e as vitórias alcançadas. Minha mãe, que pela sua perseverança e seu jeito guerreiro de ser, ensinou-me a buscar nos estudos o conhecimento para tornar o mundo um lugar melhor e sustentável, sendo um grande exemplo acadêmico. Meu pai, com seu jeito animado e diferenciado de ser, me mostrou que com os ensinamentos da escola da vida e com seu talento de encantar os seus clientes foi possível formar o filho caçula, sendo um grande exemplo de pai presente e amável.

Aos meus irmãos Nadson e Naidel, que me motivam constantemente para a realização dos meus sonhos e que independentemente do que pudesse acontecer estiveram sempre dispostos a me ajudar.

À minha namorada Bia, que está sempre ao meu lado, apoiando-me, ajudando a concretizar meus sonhos e dando forças para acreditar que tudo vai dar certo.

Aos meus familiares e amigos, pela força que sempre emanam para mim e por estarem sempre juntos em qualquer situação, ajudando-me a sempre seguir em frente e ser minha melhor versão.

Estendo minha gratidão ao professor e orientador Adriano Anunciação Oliveira, que a partir da criação de um grupo de estudo me fez buscar e despertar o interesse por esse tema presente neste projeto e cuja paciência e direcionamento me auxiliaram na realização do mesmo. Aos membros da banca, a minha gratidão por aceitarem o convite para contribuírem com esta pesquisa.

“E tudo que pedirdes em oração, crendo, recebereis.”

(Mateus 21:22)

RESUMO

O uso da renderização para uma melhor assimilação da imagem em projetos arquitetônicos requer melhoria, e , a partir dessa necessidade, este trabalho de pesquisa, veio apresentar a renderização em tempo real, na busca por atender a demanda e o mercado. Assim sendo, este trabalho de pesquisa vem apresentar esse diferencial. É com base no acompanhamento das necessidades humanas, que as novas tecnologias podem contribuir com a melhoria por meio da implementação de projetos na área da Arquitetura Urbanística, assim como em outras áreas. Este trabalho de pesquisa propõe-se a aplicação de tecnologias usualmente utilizadas em games no ofício dos profissionais da Arquitetura e Urbanismo, utilizando a renderização em tempo real a fim de otimizar os resultados de um processamento digital, conferindo também rapidez ao projeto à referida renderização. Objetiva-se melhorar a qualidade das imagens e tornar os projetos mais simples e eficientes. Para tanto, discute-se as técnicas do ray tracing e da rasterização, possibilitando o emprego da técnica de renderização em tempo real, ainda, pouco explorada na Arquitetura. A intenção também é facilitar a compreensão do usuário em relação ao projeto em discussão, permitindo a familiarização e o diálogo com aquele que compartilha o desenho. Dessa forma, desperta a relação de diálogo entre o Arquiteto e o cliente, auxiliando na construção de uma obra que satisfaça a necessidade do cliente.

Palavras-chave: Computação gráfica. Renderização. Arquitetura.

ABSTRACT

The use of rendering for a better assimilation of the image in architectural projects requires improvement, and, from this need, this research work, came to present the rendering in real time, in the search to meet the demand and the market. Therefore, this research work presents this differential. It is based on the monitoring of human needs that new technologies can contribute to improvement through the implementation of projects in the area of Urban Architecture, as well as in other areas. This research work proposes the application of technologies commonly used in games in the craft of Architecture and Urbanism professionals, using real-time rendering in order to optimize the results of a digital processing, also giving speed to the project to that rendering. The objective is to improve the quality of images and make projects simpler and more efficient. Therefore, ray tracing and rasterization techniques are discussed, enabling the use of the real-time rendering technique, still little explored in Architecture. The intention is also to facilitate the user's understanding of the project under discussion, allowing familiarization and dialogue with the person who shares the drawing. In this way, it awakens the relationship of dialogue between the Architect and the client, helping in the construction of a work that satisfies the client's needs.

Keywords: Computer graphics. rendering. Architecture

LISTA DE FIGURAS

Figura 1 – Vista arquitetônica renderizada.....	20
Figura 2 – Renderização de interior.....	22
Figura 3 – O que é Ray tracing?	25
Figura 4 – Mortal Kombat 1992.....	28
Figura 5 – Mortal Kombat 2011	29
Figura 6 – Modelo de referência do mundo real.....	32
Figura 7 – Sistema de unidade de medida no 3ds max.....	33
Figura 8 – Organização por layers do modelo no 3ds max.	33
Figura 9 – Modelo sendo editado no UVW Map.	34
Figura 10 – Alterações nos parâmetros do material.	35
Figura 11 – Material em tipo Standard.....	35
Figura 12 – Mapeamento UV do modelo tridimensional do deck.....	36
Figura 13 – Desdobramento da malha do modelo tridimensional do deck.....	37
Figura 14 – IDs do material de madeira.	37
Figura 15 – Workflow do FBX.....	38
Figura 16 – Opções de configuração do arquivo importado pelo Unreal.	39
Figura 17 – Projeto da casa importado para o Unreal.....	40
Figura 18 – Utilização do ator Lightmass importance volume.....	40
Figura 19 – <i>Build</i> em qualidade preview.....	41
Figura 20 – <i>Build</i> em qualidade high.	42
Figura 21 – Defeitos presentes na textura após o <i>Build</i>	42
Figura 22 – Tempo final de <i>Build</i> realizado pelo Swarm Agent.	43
Figura 23 – Ajustes nos níveis de luz para melhorar a luminosidade do ambiente.	44
Figura 24 – Tempo final de <i>Build</i> realizado pelo Swarm Agent.	44
Figura 25 – Ajustes da autoexposição do ator <i>post processs volume</i>	46
Figura 26 – Ajustes da luz indireta utilizando o ator Sky Light.....	47
Figura 27 – Canal 1 do objeto piso no qual foi recebido os ajustes de lightmap.....	47
Figura 28 – Ajustes feitos no objeto piso para melhorar a resolução do lightmap.....	48
Figura 29 – Ajustes com renderização em tempo real do material da cadeira.	48
Figura 30 – Imagem de referência.....	49
Figura 31 – Fachada principal da casa feita no Unreal Engine.	50

Figura 32 – Vista da fachada posterior feita no Unreal Engine.....	50
Figura 33 – Vista do interior da casa feita no Unreal Engine.....	51
Figura 34 – Vista em diagonal da cozinha.....	51

LISTA DE ABREVIATURAS E SIGLAS

AO	Ambient Occlusion
API	Application Programming Interface
CAD	Computer Aided Design
CGI	Computer Graphic Imagery
FBX	Autodesk Filmbox
GPU	Graphics Processing Units
PBR	Physically Based Rendering

SUMÁRIO

1	INTRODUÇÃO.....	13
1.1	OBJETIVO GERAL.....	15
1.2	OBJETIVOS ESPECÍFICOS	16
2	REFERENCIAL TEÓRICO.....	17
2.1	Usos da visualização arquitetônica.....	20
2.1.1	<i>Como um meio de comunicação.....</i>	21
2.1.2	<i>Como uma ferramenta de marketing</i>	21
2.1.3	<i>Breve histórico sobre o processo de desenvolvimento da visualização na Arquitetura</i>	22
2.1.4	<i>Softwares comumente utilizados</i>	24
2.1.5	<i>Tecnologias de visualização em tempo real</i>	24
2.1.6	<i>Técnicas de Ray tracing.....</i>	25
2.1.7	<i>Técnicas de rasterização</i>	26
2.1.8	<i>Renderização offline</i>	26
2.1.9	<i>Renderização em tempo real.....</i>	26
2.1.10	<i>Tecnologias de motores de jogos 3D.....</i>	27
2.1.11	<i>Soluções comerciais para a renderização em tempo real.....</i>	28
2.1.12	<i>O que essas formas de ilustração técnica transmitem e porque ela são insuficientes para publicações publicitárias de empreendimento</i>	29
3	PRODUÇÃO	31
3.1	Elaboração de um modelo arquitetônico residencial.....	32
3.1.1	<i>Mapeamento de textura - Tile.....</i>	34
3.1.2	<i>Mapeamento UV</i>	36
3.1.3	<i>Uso de ID para organizar os materiais</i>	37
3.2	<i>Workflow do FBX para Unreal.....</i>	38
3.3	Visualização do projeto após a importação para o Unreal Engine.....	39
3.3.1	<i>Utilização do Lightmass.....</i>	40
3.3.2	<i>Definindo os níveis de qualidade no mapa do projeto</i>	41
3.3.3	<i>Build do modelo</i>	41
3.4	Volume de pós processamento	43

3.4.1	<i>Luz Estática</i>	44
3.4.2	<i>Luz Estacionária</i>	45
3.4.3	<i>Luz Móvel</i>	45
3.4.4	<i>Ajustes de luz indireta e sombra</i>	46
3.4.5	<i>Ajustes de Materiais</i>	48
3.4.6	<i>Renderização final do projeto</i>	49
3.5	A renderização em tempo real como solução tendo como finalidade uma representação fotorrealista	51
4	CONSIDERAÇÕES FINAIS	53
	REFERÊNCIAS	56

1 INTRODUÇÃO

O fotorrealismo na computação gráfica pode ser definido como imagem sendo realista, ou seja, a imagem deve produzir a mesma resposta visual que a cena. A busca pelo realismo tem inspirado muitos avanços significativos em técnica de modelagem, exibição e renderização. (FERWERDA, 2003). A renderização fotorrealistas de cenas em mídia visual digital fez com que o realismo se tornasse um componente vital para área de efeitos visuais. (DINUR, 2021)

Nas visualizações arquitetônicas o fotorrealismo também está presente, são nelas que o processo de representação da arquitetura a partir de imagens realistas geradas por computador, conhecida como Computer Graphic Imagery (CGI), torna a criação de objetos tridimensionais mais próxima da realidade. (APPEL, 1968).

Os algoritmos de rasterização e de ray tracing são utilizados para a criação de uma imagem bidimensional a partir de uma cena tridimensional, configurando assim o processo de renderização. O ray tracing geralmente consegue reproduzir imagens mais fotorrealistas e com elevada riqueza de detalhes do que as cenas rasterizadas (AKENINE-MÖLLER *et al.*, 2008). Ao invés de se limitar a descobrir o que é visível a partir de um único ponto, o ray tracing pode determinar o que é visível de muitos pontos diferentes, em muitas direções diferentes.

A técnica do ray tracing traz a tona uma computação gráfica muito mais sofisticada, permitindo que o usuário visualize detalhes como desfoque de movimento, profundidade de campo, penumbras, translucidez e reflexos difusos. Por esse motivo, em especial, é que levar a renderização ao meio arquitetônico fez nascer um novo olhar de fazer arquitetura, pois é através de sua obra que o profissional dessa área se comunica com o meio que o cerca.

É no projeto arquitetônico que são elaboradas as características principais de um projeto, tais como, cômodos, quantidade de pavimentos, posicionamento de portas e janelas com suas dimensões, entre outras informações relevantes para a conclusão da obra. A técnica da renderização apresenta uma prévia aos criadores e consumidores de como determinado espaço ou construção será concluída.

Nesse sentido, percebemos que é importante gerar visualizações para que possíveis clientes não técnicos compreendam o projeto. Isto posto, a renderização permite o efeito da familiarização no cliente, facilitando o entendimento da criação e permitindo que o usuário auxilie o criador do projeto em seu desenvolvimento, pois o caráter realista das imagens facilita a compreensão e o engajamento na obra.

Historicamente, essas visualizações arquitetônicas eram apenas ilustrações, mas,

como a necessidade de obter uma imagem de um projeto no qual ainda não estava construído, era um interesse de muitos desses clientes, surgiu-se a necessidade de funcionalizar e/ou popularizar os projetos. E é nesse contexto, que a renderização ganhou notoriedade.

Nesse cenário, com o avanço da tecnologia propriamente dita e da tecnologia na Arquitetura, computadores pessoais foram se tornando cada vez mais capazes de gerar representações realistas de projetos arquitetônicos.

Até então os projetos estavam cada vez mais realista, embora, se apresentavam onerosos e inacessíveis. Os softwares empregam técnicas custosas de rasterização e ray tracing como exemplo, Autodesk 3ds max, Autodesk Maya, Vray, entre outros. Hoje, com o avanços das demandas sociais, das técnicas arquitetônicas e das tecnologias para construções, torna-se necessário empregar originalidade aos projetos, conferindo realística e agilidade ao mercado, tais como o uso do ray tracing, como observado anteriormente.

Com o avanço da tecnologia gráfica advieram diversos algoritmos de visualização de superfícies, mas nenhum se mostrava tão eficiente quanto o Ray Tracing, podendo ser compreendido da seguinte forma:

“O algoritmo se baseia numa idéia muito simples: um observador se senta em frente a uma tela plana transparente. De seus olhos partem diversos “raios visuais” que vão atravessar os pontos da tela e bater nos objetos tridimensionais, que foram definidos utilizando-se alguma técnica de modelagem. Pintamos, então, o ponto da tela que foi atravessado pelo raio com a cor do objeto que foi atingido por este. Esta é a forma mais simples de se apresentar o algoritmo de Ray Tracing.”(SILVA, 1994, p. 2)

Desse modo, apesar de simples, o ray tracing se apresenta como uma ótima alternativa ao mercado e surgiu com a necessidade de conferir verossimilhança e alto grau de realismo, no qual pode ser assimilado da seguinte forma:

“Os primeiros trabalhos sobre o uso da técnica de traçado de raios para renderização em tempo real datam da década de 80 [3] e surgiram da necessidade de sintetizar imagens mais realistas, diante das limitações do modelo de computação local, conhecido como rasterização. Neste modelo, muito esforço de programação é necessário para se incluir simulações de sombra, reflexão e refração que, por outro lado, são produtos diretos da traçado de raios.” (NERY, 2010, p. 4)

Complementarmente, a rasterização, que, em síntese, é um processo de amostragem de um modelo 3D em 2D utilizando linhas e pontos, apresenta-se como uma técnica paralela ao ray tracing, que, também se manifesta pela emissão de raios em determinada cena, ou, no caso deste relatório, emissão de raios sob o projeto ou esboço de uma obra arquitetônica. É uma

técnica que produz uma imagem vista de um único ponto de vista, diferentemente do ray tracing, quando podemos observar uma imagem a partir de várias perspectivas (SANTOS *et al.*, 2018).

Por isso a técnica ray tracing ganhou tanta adesão nos últimos anos, devido a sua capacidade de rastrear raios de várias fontes/ângulos possíveis, característica que torna possível simular como a luz se espalha no mundo real com muito mais precisão do que é possível com a rasterização.

Dito isto, o objeto do trabalho não é discutir qual das técnicas é a melhor, mas entender que ambas se complementam, evidenciando seus pontos fortes e seus desafios. E que no uso ora proposto, qual seja, utilização da tecnologia de *video games* nas obras arquitetônicas, utilizando o Unreal Engine, a rasterização ganha maior notoriedade por ser mais funcional ao plano de ação que está sendo apresentado. No entanto, isso não quer dizer que o ray tracing não será utilizado. A proposta é utilizar as potencialidades das duas técnicas para um resultado otimizado e usar a rasterização para um resultado em tempo real, diferentemente do ray tracing que é mais utilizado em renderizações *offline*. A popularização dessas novas técnicas, resultante da grande necessidade de sintetizar imagens realistas, acima mencionada, permitiu uma rápida assimilação por diversas áreas. Estima-se que o número de usuários que utilizam software de renderização 3D seja de 2,1 milhões com base no uso do software e que o crescimento de mercado do ray tracing possui uma taxa de crescimento anual composta de 5% e deve ser de quase um quarto de bilhão de dólares em 2023 (PEDDIE, 2019).

Em paralelo, nos últimos 10 anos, tecnologias de *video games* começaram a ser usadas para criar visualizações impactantes e rápidas. Assim sendo, se faz necessário que pesquisas avancem nessa perspectiva e contribua com a ciência e se insira no mercado de trabalho de forma mais rápida, otimizando a arquitetura convencional.

Portanto, este estudo se justifica e torna-se necessário, tendo em vista o mercado de trabalho exigir uma tecnologia mais ágil e eficiente, a partir do fato de que a demanda da área da Arquitetura e Urbanismo tem crescido e exigido das tecnologias uma melhora nos seus produtos. Assim, este relatório técnico buscou traçar os seguintes objetivos:

1.1 OBJETIVO GERAL

O objetivo desse projeto é oferecer caminhos para explorar as potencialidades do desenvolvimento de uma visualização arquitetônica fotorrealista aplicada ao mercado da Arquitetura, de modo a apresentar as capacidades atuais e explorar limites dos mecanismos de técnicas

de renderização em tempo real.

1.2 OBJETIVOS ESPECÍFICOS

Compreender as potencialidades específicas das técnicas de visualização fotorrealista baseadas em tecnologia de video games, visando a divulgação da tecnologia empregada para que novos trabalhos possam ser exibidos aos clientes de forma mais satisfatória.

Empregar a técnica de renderização em tempo real de forma mais eficaz e eficiente em projetos arquitetônicos.

Desenvolver um ambiente arquitetônico com alto grau de realismo, observando as limitações e potencialidade dos sistemas atuais.

2 REFERENCIAL TEÓRICO

Com o objetivo de realizar investigação sobre o tema deste projeto, foi necessário recorrer a pesquisa sobre os autores que trazem na sua literatura o estudo de imagem de forma geral, aprofundando-se em temas mais específicos, tais como utilização das imagens fotorrealísticas em games, o empréstimo desta tecnologia para uso na arquitetura e, mais especificamente, a literatura referente aos softwares ora analisados.

Dessa forma, o objetivo desse capítulo é apresentar as premissas basilares para a compreensão da técnica abordada nesse trabalho, debruçando-se no que já está sendo estudado nesse campo acadêmico-teórico e no que já está sendo implantado na Arquitetura usual, ou seja, nos projetos já em curso.

Inicialmente, é necessário compreender que a imagem é um meio de comunicação substancial para arquitetura, sendo por meio da representação dos projetos que a arquitetura dialoga com a sociedade. As imagens são divididas em dois domínios, uma é caracterizada por representações visuais como desenhos, pinturas, fotografia, imagens cinematográficas e televisivas. O outro domínio é o imaterial, ou seja, a manifestação das imagens na mente humana. É neste domínio, que as imagens são apresentadas como fantasias, imaginações, visões, ou seja, representações mentais. Os dois domínios não podem existir separadamente, pois estão interligados desde a sua origem (SANTAELLA; NÖTH, 2020).

Com base nessa definição, podemos compreender de forma mais satisfatória que os projetos arquitetônicos são como uma imagem mentalizada pelo arquiteto, no qual precisa ser representada visualmente, para que seus clientes possam compreender o que ele quer transmitir, criando maneiras de visualizar o projeto antecipado da construção. É neste momento criativo que o uso de tecnologias mais avançadas se insere na Arquitetura, permitindo uma melhor comunicação entre as partes da criação. As inovações tecnológicas na Informática não permitiram exclusivamente a automatização dos processos usualmente manuais, mas também um ganho de produtividade e de qualidade ao produto final, possibilitando uma variedade de produtos e serviços, fomento a competitividade e ânsia pela melhoria dos serviços (STEINGRÄBER, 2003).

Com o passar dos anos, as imagens sintéticas desenvolvidas por computadores, vêm sendo cada vez mais utilizadas para criar as primeiras impressões visuais com o objetivo de antecipar o que será construído. As imagens sintéticas, podem ser denominadas de renderização digital, no qual pode ser considerado como o processo finalizado do processamento digital executado por programas de modelagem Bidimensional (2D) ou Tridimensional (3D) (MORAIS,

2016, p. 174).

Quando a ideia é transferida para o físico, o planejamento ganha forma e materialidade, tal como era realizada no passado com as maquetes, hoje é compartilhado por meio da maquete eletrônica. O termo maquete eletrônica designa um tipo de imagem criada em uma ambientação computacional no qual possui o objetivo de demonstrar um projeto arquitetônico utilizando uma modelagem tridimensional (BRAIDA *et al.*, 2017).

É neste contexto, que o uso do fotorrealismo em tempo real ganha notoriedade, pois, além de se valer de uma tecnologia instantânea, resulta em um trabalho eficiente em seus aspectos, a partir do detalhamento do produto. Sob essa ótica, é a partir do detalhe que uma construção ganha identidade própria e, conseqüentemente, o poder de representar quem a possui (BROOKER; STONE, 2014).

Assim, o detalhamento trazido com o uso da tecnologia, assim como o caráter imediato da disponibilização do produto final, são direcionamentos dos novos projetos arquitetônicos, sendo as obras um produto da mente do Arquiteto e da relação com seu entorno (RIGHETTO, 2005). Assim sendo, pode ser melhor compreendido da seguinte forma:

“A computação gráfica permitiu aos arquitetos inovarem nas representações, bem como o modelo virtual veio integrar conceitos que os arquitetos vinham buscando: ver por simulação o interior e o exterior do edifício, a relação deste com o entorno, o estudo do emprego de materiais com a colocação de texturas, jogo de luzes e sombras, superposição, fotorrealismo e tantas outras possibilidades.”(RIGHETTO, 2005, p. 425)

Somado a isso, o empréstimo da tecnologia de *video games* confere realidade e precisão, tal como a visualizada nos jogos, os quais precisam dessa eficácia, por se tratar de uma interface que precisa de renderização em tempo real, para que um *frame* seja construído/jogado ao espectador imediatamente após o outro, permitindo a comunicação instantânea.

Nesse sentido, podemos citar os “Game Engines”, também conhecidos como ‘Motores de Jogos’, que são coleções de módulos de códigos de simulação feitos para otimizar o desenvolvimento dos jogos os quais podem ser úteis para projetos arquitetônicos. O mecanismo de renderização é a parte mais valiosa do jogo, pois nele é incorporado o código complexo que são necessários para identificar e renderizar a visão do jogador de um modelo 3D do ambiente. A análise dessa tecnologia em ação é o ponto principal deste relatório na medida em que, percebendo sua eficácia no campo dos *video games*, criou-se a ideia de que a mesma eficácia poderia ser vista em projetos de outras áreas (LEWIS; JACOBSON, 2002).

Todavia, ainda existem lacunas a serem sanadas neste ofício, uma vez que não há o aprofundamento necessário nas tecnologias já existentes, fato que dificulta a produção das imagens.(ZANCANELI *et al.*, 2019). Alguns *renders* ainda não são gratuitos e possuem interface de difícil interação. Desse modo, pode ser melhor compreendido da seguinte forma:

"Mesmo com o advento das tecnologias computacionais podendo ser um facilitador para a geração de imagens de apresentação, por vezes, a falta de conhecimento das ferramentas, assim como a falta de habilidade com os meios digitais de representação, dificulta a produção das imagens. No caso dos projetos de interiores, isso pode ter um efeito prejudicial ainda maior, uma vez que são projetos com alto nível de detalhe onde cada parâmetro de iluminação ou de materiais configurado de maneira equivocada pode causar um efeito indesejado e não transmitir as ideias do arquiteto da forma esperada."(SOARES, 2007, p. 7)

Isto posto, propõe-se um aprofundamento na análise de técnicas de renderização, com a finalidade de alcançar a funcionalidade do espaço e uma estética cada vez mais real

O ray tracing, uma das técnicas aqui estudadas, é um algoritmo composto por conceitos básicos da Física Óptica, no qual tem como principal característica a alta qualidade visual a um custo computacional bastante elevado e que mantém aplicações de uso offline (MELO, 2012). Com o avanço do poder gráfico das Graphics Processing Units (GPU), este algoritmo se tornou viável para ser usado em aplicações em tempo real, devido ao fato de possuir características de paralelização, permitindo o uso nos projetos arquitetônicos ora explanados, e viabilizando o uso por um público mais geral.

Dessa forma, a evolução da utilização do ray tracing se manifesta como um desenho histórico que caminha em paralelo com a rasterização, método também bastante utilizado na computação gráfica. Nesse ambiente a rasterização, o ray tracing e até o path tracing, recente inovação na computação gráfica que permite uma precisão ainda maior do que a do ray tracing, complementam-se com uma finalidade em comum, a renderização, que aqui, sob a ótica do uso das tecnologias de *video games*, funciona como vanguarda nos estudos arquitetônicos com grande potencial para o êxito.

Nesse contexto, com o avanço das tecnologia de informação, é possível perceber que disciplinas como computação gráfica já estão inseridas nos cursos de Arquitetura e Urbanismo, com o objetivo de capacitar os alunos para o mercado de trabalho (PINHAL, 2017). Isso mostra que a multidisciplinaridade é uma ferramenta valorosa para o avanço das tecnologias em diversas áreas, ainda mais quando as tecnologias atuais já não mais atendem as necessidades (SOARES, 2007).

Isto posto, surge a necessidade de buscar instrumentos e estudos cada vez mais qualificados para auxiliar no processo de criação e execução. Sobre esse tema, arremata Soares (2007, p.7):

“Os métodos tradicionais de desenhos para produtos industriais não conseguem mais atender à velocidade e precisão exigidas. É neste âmbito que surgem as condições adequadas para o aparecimento e aplicação de novas tecnologias computacionais ao processo de projeto”.

Portanto, é importante que haja o aprofundamento no “fazer arquitetura” permitindo detalhar todo o percurso percorrido até unindo aos benefícios de novas tecnologias, momento no qual está sendo proposto uma “virada de chave”, que, apesar de ter sido natural e eficiente, quando falamos no uso da renderização na Arquitetura, necessita que seja realizado de forma analítica e original, pois a todo momento novas técnicas e formas de utilização surgem, tal como a que está sendo proposta neste relatório.

2.1 Usos da visualização arquitetônica

No cenário atual, os arquitetos utilizam de imagens renderizadas por softwares tradicionais para apresentarem seus projetos aos seus clientes. A Figura 1, a seguir, apresenta como são essas imagens renderizadas para que os clientes possam ter uma melhor avaliação do ambiente a ser construído, tornando desenhos complexos mais compreensíveis:

Figura 1 – Vista arquitetônica renderizada



Fonte: Sítio Virtual.¹

2.1.1 Como um meio de comunicação

Um dos principais objetivos da visualização arquitetônica é facilitar a comunicação entre os arquitetos e clientes. Estes profissionais da arquitetura trabalham com uma variedade de dados e informações do desenho técnico bidimensional, no quais geralmente os clientes não possuem conhecimento suficiente sobre o assunto, assim sendo, teriam que aprender alguns conceitos e habilidades para interpretar os projetos arquitetônicos. Mesmo o cliente possuindo alguns conceitos, a sua comunicação não estará em um mesmo nível do profissional da arquitetura. A comunicação verbal é muito complexa para comunicar com clareza as relações espaciais. (CHEN, 2004)

Ainda é bastante utilizado a comunicação visual tradicional com desenhos técnicos, no qual, requer a geração de imagens mentais para tentar compreender melhor o projeto. Além do mais, a comunicação pode ser falha devido o arquiteto e o cliente estarem criando imagens mentais distintas. Na imagem exibida logo acima, por exemplo, observa-se a representação de uma zona urbana, de forma detalhada e realista, tão real que na imaginação do cliente já está sendo introjetado como a obra já executada.

A partir disso, pode-se depreender que além de uma estratégia de compartilhamento da imagem, a renderização funciona como uma estratégia de marketing, pois mostrar a imagem de um produto tão similar ao real leva o espectador a já se imaginar obtendo-o, isso possibilita a venda do produto pelo sentimento de pertencimento do cliente.

2.1.2 Como uma ferramenta de marketing

As visualizações arquitetônicas de marketing englobam imagens renderizadas em alta definição do interior quanto exterior de residências e até mesmo edifícios. Essas imagens são produzidas para um público-alvo específico com o objetivo de criar uma resposta emocional à imagem. Os interiores geralmente apresentam mobília luxuosa e detém vários posicionamento de luz importantes para a cena, em alguns casos, desenvolver uma ilusão de publicidade se torna mais importante do que mostrar com precisão a realidade.

Quanto à renderização, a ideia principal é formular um *render* do projeto como um todo, produzindo uma imagem realista do resultado que se pretende. Isso facilita o que antes era uma lacuna a ser preenchida na arquitetura, qual seja, tornar um projeto realista ao olhos dos clientes, eventuais investidores, avaliadores, e até mesmo para emissão de licenças pelos

Figura 2 – Renderização de interior.



Fonte: Diandra Tauil.

órgãos fiscalizatórios. Assim, como mencionado anteriormente, essa técnica permite que um profissional da arquitetura elabore animações ou imagens bidimensionais com a finalidade e compartilhar todos as particularidades e atributos que precisam ser ser incorporados no design final.

2.1.3 Breve histórico sobre o processo de desenvolvimento da visualização na Arquitetura

No passado, o desenho arquitetônico já era considerado como uma ferramenta essencial na vida do arquiteto, pois para representar graficamente um ambiente era necessário materiais como, lápis, esquadros, régua, escalímetros entre outros. Para que o desenho ficasse mais realistas os profissionais precisavam possuir habilidades para criar perspectivas, vistas e noções de escala no desenvolvimento do projeto, gerando um retrato do corpo social durante os séculos (SONTAG, 2004).

Desse modo, percebe-se que a história da civilização humana se harmoniza com a história da representação gráfica, uma vez que desenhos humanos são realizados desde os períodos mais remotos, vide a arte rupestre. Assim sendo, pode ser melhor compreendido da seguinte forma:

“Uma sociedade se torna “moderna” quando uma de suas atividades principais

consiste em produzir e consumir imagens, quando imagens que têm poderes excepcionais para determinar nossas necessidades em relação à realidade e são, elas mesmas, cobiçados substitutos da experiência em primeira mão se tornam indispensáveis para a saúde da economia, para a estabilidade do corpo social e para a busca da felicidade privada.” (SONTAG, 2004, p. 169)

Segundo Santaella & Nöth(2020), as “[...] imagens têm sido meios de expressão da cultura humana desde as pinturas pré-históricas das cavernas, milênios antes do aparecimento do registro da palavra pela escritura”. Desse modo, os arquitetos utilizam suas amostragens como meio de comunicação com o mundo ao redor. Portanto, é necessário que haja sempre a busca pela inovação nessa área, pois, as tecnologias vêm acompanhadas aos anseios da sociedade.

Assim, a fotografia, importante meio de representação de imagens na humanidade, manifesta-se como testemunhos humanos, vez que, quando duvida-se de algo, logo é solucionado quando estamos diante de uma fotografia, que se manifesta como um retrato do tempo (SONTAG, 2004).

Com o passar dos anos, a tecnologia trouxe ferramentas que se tornaram imprescindíveis na vida do profissional da arquitetura, o computador foi uma das mais importantes no qual é responsável pela maior parte da produção de imagens pós-fotográficas. Atualmente, para além da fotografia, tem-se a computação gráfica que contribui com a ideia, podendo ser melhor definida assim: Zancanelli (2019, p. 31)

“Um projeto consiste em uma ideia, uma imagem mental produzida por quem projeta, ideia essa que fora baseada no repertório pessoal de imagens visuais do arquiteto ou designer e que precisa ser externada na forma de uma imagem visual que seja a síntese do processo, a fim de que o executor ou cliente compreenda e visualize o projeto. Dessa forma, as imagens visuais mostram-se como elementos imprescindíveis para o processo de projeto, uma vez que traduzem as imagens mentais, que é o projeto em si.”

Dito isso, a Arquitetura absorveu essa ideia e unificou todos os mecanismos de compartilhamento de um projeto. No relatório ora apresentado propomos o uso otimizado da computação gráfica e suas particularidades evidenciando que os programas computacionais estão cada vez mais poderosos e sofisticados, permitindo assim a criação de universos virtuais no quais podem apresentar-se como tal, mas também pode fazer falsificações com imagens aparentemente real. A instalação de imagens interativas podem permitir a imersão parcial ou total do espectador em um universo virtual (JOLY, 1996).

2.1.4 Softwares comumente utilizados

Uma grande parte de profissionais arquitetos e designers utilizam a tecnologia computacional como ferramenta de apresentação e representação de seus projetos. Dessa forma, essas ferramentas possuem características facilitadoras para esses profissionais inseridos no mercado de trabalho digital, no qual pode se tornar uma representação gráfica mais precisa quando utilizam softwares de Computer Aided Design (CAD), que incluem também recursos de renderização para obter uma imagem fotorrealística para as apresentações de projetos.

Esses softwares como, 3ds max, SketchUp, Revit e Blender, oferecem possibilidades de compartilhamento rápido e maior interação com o projeto em diferentes fases. É possível perceber também o uso mais frequente de imagens digitais fotorrealistas na fase de divulgação de projetos aos cliente, tanto para pessoas físicas quanto para empresas, construtoras e até mesmo laboratórios de ateliê em universidades de arquitetura.

A partir da década de 1980, com o surgimento dos softwares **Computer-Aided Design (CAD)** o computador passa a se tornar mais presente nos escritórios de arquitetura como instrumento de desenho. (SANTOS, 2005). As tecnologias CAD colaboram na visualização de projetos e fabricação de objetos em ambientes reais e virtuais a partir de desenhos com base em vetores que estariam inserido em espaços bidimensionais ou na modelagem e na animação de figuras do espaço tridimensional. (CHING, 2000).

2.1.5 Tecnologias de visualização em tempo real

Muitas variedades de algoritmos de renderização foram explorados para concretização deste relatório, com o objetivo de captar as peculiaridades de cada método. A partir dessa análise, depreende-se que os softwares utilizado para renderização podem empregar diversas técnicas diferentes para produzir uma imagem final. Realizar cada traçado da partícula de luz em uma cena é um processo bastante complexo no qual requer bastante tempo para chegar no resultado pretendido.

Isto posto, surgiram as principais técnicas de renderização da imagem que tornam esse processo mais eficiente:

2.1.6 Técnicas de Ray tracing

Com o avanço da tecnologia, o meio arquitetônico já se vale de estratégias ou instrumentos de outras áreas para auxílio em seus projetos. Um bom exemplo disso é o uso do 3DS Max, SketchUp, Revit, Blender, que são utilizados para construir um projeto, e o Lumion, V-ray, Corona para renderizarem o projeto, ou seja, a produção da imagem realista por meio do projeto 3D já construído.

A partir do uso do Ray Tracing, por exemplo, há a possibilidade de atribuir ao render traçados de luz e sombra muito mais realistas (NERY, 2010), tal qual utilizado nos filmes e *games*, gerando em uma melhor translucidez e a dispersão de uma forma mais real. Esta é uma tecnologia disponível para quase todos os renders, tal como o Unreal Engine, que dão o suporte para a aplicação da tecnologia, ou seja, é uma realidade possível para este novo cenário, chegando ao alcance de muitas áreas, inclusive à arquitetura e conferindo realidade e precisão às imagens, tal como é empregado nos *video games* (Figura 3).

Figura 3 – O que é Ray tracing?



Fonte: Sitio virtual.²

A realidade mencionada acima, ao passo que se familiariza com o posto inovador que a Arquitetura ocupada, contrasta com os métodos antes aplicados, tendo em vista a alta tecnologia empregada e o uso revolucionário de uma forma de compartilhamento moderna, eficiente, detalhista e interativa. Propõe-se aqui um projeto além da renderização, alcançando tecnologias mais refinadas e elaboradas.

² <https://blogdaengenharia.com/secoes/colunistas-blog-da-engenharia/o-que-e-ray-tracing/>

2.1.7 *Técnicas de rasterização*

A Rasterização é uma técnica de transmutação da representação vetorial para a matricial. Esse método utilizado na computação gráfica possibilita a realização da conversão de um projeto 3D em uma representação 2D, capaz de ser replicada em um dispositivo *raster*, muito mais simples e de fácil visualização (LOPES, 2013).

É uma técnica mais simplificada que o Ray tracing e utilizada em projetos que não exijam muito detalhismos.

2.1.8 *Renderização offline*

A renderização offline é utilizada em momentos nos quais não há tanta necessidade de velocidade, ou pelo menos não se trata de uma situação onde é necessário renderizar de forma instantânea, como exemplo nos filmes, produto que exige um processo de criação de meses, possibilitando uma margem de tempo maior para a criação. Como bem afirma Santos (2018, p. 26).

"Rendering off-line é o tipo de rendering comumente usado em produções de animação 3D tanto em longas, quanto em curtas e produções para televisão, sendo também utilizado em outras áreas como: efeitos visuais, visualização arquitetônica, entre outras. No rendering off-line podem-se levar horas ou até dias para gerar uma única imagem ou frame de animação, por isso, grande parte das produções faz uso de render farms, isto é, uma rede de computadores conectados entre si dedicados apenas ao processo de renderização."

Nessa técnica não há a variável da imprevisibilidade como na renderização em tempo real, por esse motivo também é chamada de “pré-renderização”, em casos como esse, o cálculo *render* é feito pela CPU (processador) principal com o apoio da memória do computador e do processador gráfico da placa de vídeo, este sendo facultativo seu uso.

Como exemplo da aplicabilidade da renderização offline podemos citar a renderização de vídeo (Adobe Premiere, Final Cut e Sony Vegas), e renderização de imagem 3D (SketchUp, after effects, Photoshop e 3Ds Max).

2.1.9 *Renderização em tempo real*

A renderização em tempo real é comumente aplicada em *games* e gráficos interativos, em virtude do caráter frenético desse tipo pipeline gráfica, calculando imagens a partir de algoritmos 3D numa frequência acelerada. Para tanto, necessita-se de um hardware gráfico

para viabilizar o processamento ágil das imagens gráficas, a partir disso, depreende-se que a característica instantânea surgiu como uma alternativa viável para acelerar o processo de rendering sem perder a qualidade do produto (SANTOS *et al.*, 2018).

Em 2018, empresa a NVIDIA, multinacional da área da computação gráfica, inaugurou uma inovadora geração de placas de vídeo, intitulada RTX, sendo núcleos específicos para calcular ray tracing (NVIDIA, 2018), incentivando o uso dessa técnica, até então restrita aos *games*. Ademais, conforme consta no sítio virtual da (NVIDIA, 2018), os núcleos de ray tracing conferem às placas de vídeo possibilidade infindas, tais como a utilização do ray tracing em tempo real, importante no contexto dos jogos de video game.

A partir dessa inovação no ray tracing, podemos analisar uma forma mais refinada de fazer os jogos atualmente, nos quais existe renderização instantânea, possibilitando uma qualidade que capta o usuário pela esmero da criação. Para (AKENINE-MO *et al.*, 2018), o rendering em tempo real dá ao usuário uma sensação de pertencimento, de conexão com o espaço 3D, pois a experiência é em tempo real e de fato bem próxima à realidade física, nas dimensões de tempo e espaço.

2.1.10 Tecnologias de motores de jogos 3D

A fim de auxiliar e tornar mais prático a criação de *games*, foram desenvolvidas procedimentos que mitigaram algumas etapas do processamento dos jogos, facilitando o processo. No contexto dessa inovação surgiu a Application Programming Interface (API) - Interface de Programação de Aplicativos - que funcionam como bibliotecas na linguagem de programação gráfica que possibilitam a redução do volume de funções que o criador do projeto tinha que realizar para formular um jogo (OLIVEIRA, 2013).

Concomitantemente, advieram os softwares que eram formados por espaços de desenvolvimento, quase como uma oficina onde se tinha os instrumentos à palma da mão, nos quais os jogos são produzidos em alto nível, utilizando poucos códigos para conclusão do jogo (OLIVEIRA, 2013).

Nesse contexto, os motores de jogos funcionam como bibliotecas de desenvolvimento, supramencionadas, dando apoio à linguagem e à API gráfica. Assim, atuam na estruturação do jogo, no gerenciamento das imagens, no processamento da entrada de dados, dentre outras funções (GOMES; PAMPLONA, 2005).

A principal funcionalidade de um motor de jogo 3D é oferecer um conjunto de

soluções para uma variedade de funções, como simulação física, animação, mecanismos de renderização, efeitos de iluminação, etc. Para que fosse possível obter esses softwares era necessário um alto custo de aquisição devido a sua licença de uso.

Mas foi em 2009, que essa realidade mudou, a empresa Epic Games lançou o Unreal Development Kit. Esse kit engloba aos desenvolvedores a possibilidade de licenciar grandes funcionalidades do motor de jogo por custos extremamente baixos. Nos próximos anos, outras empresas como a Unity seguiu o mesmo exemplo.

Por causa da concorrência e a grande velocidade nos avanços da tecnologia da GPU, os recursos de renderização dos jogos aumentaram notavelmente em sua qualidade, como pode ser observado nas Figuras 4 e 5, a seguir:

Figura 4 – Mortal Kombat 1992.



Fonte: Motor Midway Y Unit.³

2.1.11 Soluções comerciais para a renderização em tempo real

Os sistemas de renderização em tempo real são baseados em processamento paralelo por meio de uma unidade de processamento gráfico GPU. Com a evolução da computação gráfica, instrumentalizou-se a maneira de fazer renderizações em tempo real, haja vista que, para tal

³ <https://www.techtudo.com.br/noticias/2015/07/mortal-kombat-confirma-evolucao-de-grafico-dos-titulos-da-serie.ghml>

⁴ <https://criticalhits.com.br/games/mortal-kombat-9-todos-os-combos-de-cada-personagem/>

Figura 5 – Mortal Kombat 2011



Fonte: Motor Unreal Engine 3.⁴

tipo de renderização é necessário um *hardware* com alto custo computacional. A partir desses avanços, muitas foram as áreas que aderiram à computação gráfica em seus empreendimentos, propõe-se então um passo maior, porque não utilizar a renderização em tempo real, típica dos *video games* para otimizar os ganhos no ramo da arquitetura?

2.1.12 O que essas formas de ilustração técnica transmitem e porque ela são insuficientes para publicações publicitárias de empreendimento

Com o avanço da tecnologia, os escritórios internacionais de arquitetura procuram aumentar a eficiência nas suas tarefas possibilitando assim uma maior exploração criativa utilizando as tecnologias mais recentes. A renderização em 3D tem como objetivo de solucionar problemas de visualização de ideias do arquiteto para o cliente no qual este não consegue compreendê-las antes de iniciar a construção de um projeto arquitetônico, representando o projeto de uma forma mais precisa e possuindo qualidade impactante antes de construir.

Para compreendermos melhor sobre as formas de representação dos projetos arquitetônicos Baratto (2016) afirma que

“Renders, ou imagens computadorizadas fotorrealistas, por outro lado, tratam de apresentar simulações de arquiteturas que ainda não foram construídas. Através de modelagem 3D, renderização e edição, é possível fazer ver aquilo que até então era exclusivo da mente de um arquiteto ou grupo de pessoas. E isso vem sendo feito com incrível “realismo” a ponto de confundir a mente do espectador que, incrédulo diante das imagens, se pergunta: mas isso é de verdade?”

Desse modo, mesmo representando uma arquitetura de forma mais clara e de fácil entendimento, as imagens digitais fotorrealísticas são cada vez mais questionadas a respeito o

seu aspecto estático, assim sendo, para se ter uma noção de espaço espacial o observador quanto estar em movimento passa a ter uma condição básica de compreensão de espaço (BOAS, 2005).

A cinematografia tem um papel importante para a possível solução da estaticidade das imagens, no qual alega que a compreensão do espaço não pode ser considerada a mesma do que ser vivenciado em um ambiente físico (ZEVI, 1996). Por esse motivo a sensação de espanto no usuário é tão positiva, pois este se ver imerso na realidade, ainda que virtual (BARATTO, 2016).

Assim sendo, o uso tecnológico dos computadores proporcionou a reprodução de vídeos de projetos arquitetônicos, no qual permite uma melhor compreensão das informações que estão presentes nos espaços desses projetos de forma mais dinâmica. Seu posicionamento a respeito dos vídeos é que a experiência física do espaço é a forma mais importante para compreender e ser parte dele, no qual as outras ferramentas continuam a serem necessárias.

3 PRODUÇÃO

A partir da definição dos objetivos do produto, a produção do projeto tem como base avaliar as potencialidades exigentes para chegar em visualização fotorrealística de um projeto arquitetônico utilizando técnicas de renderização a partir de tecnologia de game.

O projeto ora apresentado se manifesta pela tentativa de tornar as criações arquitetônicas mais próximas à realidade e agilizar o processo. Nos *games*, a renderização em tempo real é a tônica da produção, uma vez que, como a utilização do jogo é feita concomitante à renderização, é essencial que a renderização seja feita instantaneamente.

Isso exige um *hardware* com requisitos para rodar o motor de jogo, podendo utilizar uma placa gráfica com uma quantidade de VRAM e taxa de bits consideráveis para melhorar o desempenho, e que o utilizador tenha a expertise para o manejo do *software*. Com o avanço da tecnologia da informação, a técnica não está mais restrita nas mãos dos profissionais de uma área em específico.

O *software* aqui utilizado, no caso, o Unreal Engine 4.26.3, tem um interface intuitiva, a engine oferece todos os recursos de forma gratuita, como bibliotecas, materiais e suporte baseado na comunidade, a engine possui suporte em português, ou seja, apesar de existirem desvantagens, os pontos positivos permitem a adesão de usuários de outras áreas.

A ideia central desta produção é construir um projeto residencial utilizando o software acima mencionado, valendo-se do *layout* interativo e instrumentalizado a fim de produzir uma reprodução fotorrealista do projeto, expondo os *gaps* da plataforma, as dificuldades e os acertos da produção, e o manejo do projeto gráfico na plataforma.

O projeto residencial foi exportado do 3Ds Max no formato FBX, sendo essa considerada uma das principais formas de trocar informações entre os programas de 3D pertencentes à empresa AUTODESK, proprietária da forma de compartilhamento FBX. (SANTOS *et al.*, 2018)

Programas de outras empresas similares também detém suporte para importar arquivos em fbx como é o caso da Unreal Engine 4.6.3, que foi utilizado para esse projeto. Isto posto, tem-se que o fbx é um formato popular para trabalhos em 3D (SANTOS *et al.*, 2018), podendo conter geometria dos modelos, hierarquia de bones, deformadores como *skin* e *morph targets*, materiais, animações, câmeras e luzes, perfazendo uma engenharia gráfica facilitadora no que cerne a troca de informação entre programas distintos.

3.1 Elaboração de um modelo arquitetônico residencial

A etapa de elaboração da casa foi desenvolvida dentro do software 3DS MAX, com base em um modelo de referência, de casa do mundo real para servir de inspiração para o desenvolvimento e posteriormente exportado para o motor Unreal. Para que o projeto se torne o mais realista possível foi utilizado texturas de alta qualidade. Para otimizar o tempo desse projeto foi utilizado de uma estrutura pré modelada e com alguns assets e texturas completas de projetos realizados anteriormente. Utilizamos um modelo de referência para ajudar a aproximar o projeto da realidade, como observado na Figura 6, a seguir.

Figura 6 – Modelo de referência do mundo real.

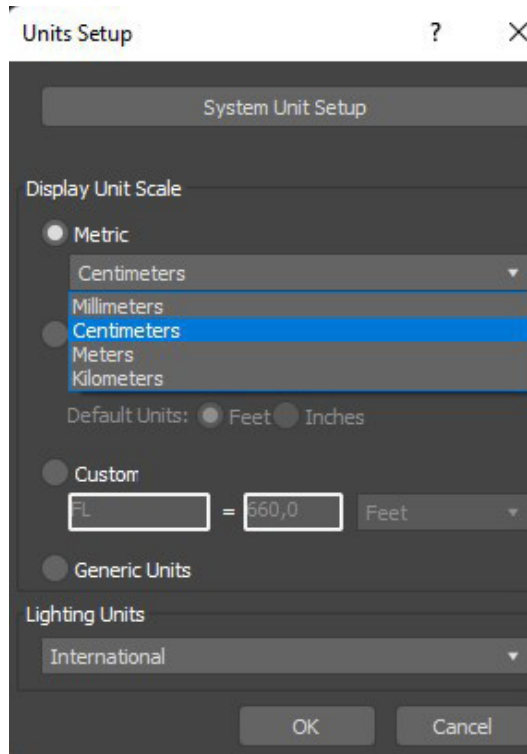


Fonte: Marcelo Donadussi.¹

O primeiro procedimento a ser feito é confirmar se o sistema de medidas do arquivo é o mesmo sistema de medidas do Unreal, isso é importante para que o arquivo feito no 3ds max seja exportado para Unreal na mesma escala. O Unreal usa o sistema de medidas em centímetros, então para confirmarmos isso dentro do software é necessário ajustarmos as medidas do projeto por completo para centímetros , como mostra a Figura 7, a seguir.

¹ https://www.archdaily.com/926128/c5-house-martin-arquitetura-plus-engenharia?ad_medium=gallery

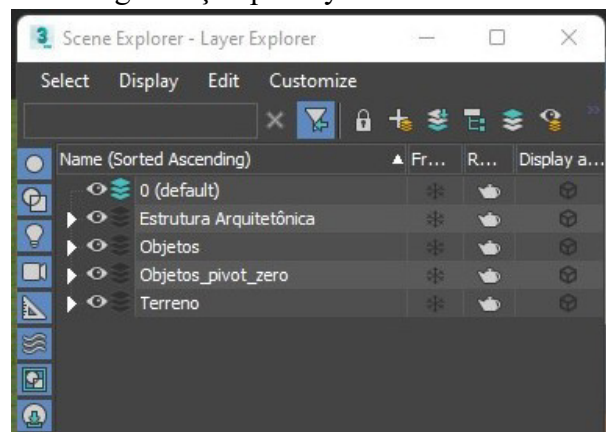
Figura 7 – Sistema de unidade de medida no 3ds max.



Fonte: Elaborada pelo autor.

Para que o projeto pudesse ser melhor organizado e descobrir o que já foi realizado durante a elaboração da casa, foi adotado uma organização por layers no qual a casa ficou dividida em quatro principais layers quais sejam: terreno, estrutura arquitetônica, objetos e objeto pivot zero. Este último foi feito pra facilitar a localização dos objetos quando forem importados posteriormente pela Unreal, ficando localizado no ponto zero do projeto, como apresentado na Figura 8, a seguir.

Figura 8 – Organização por layers do modelo no 3ds max.

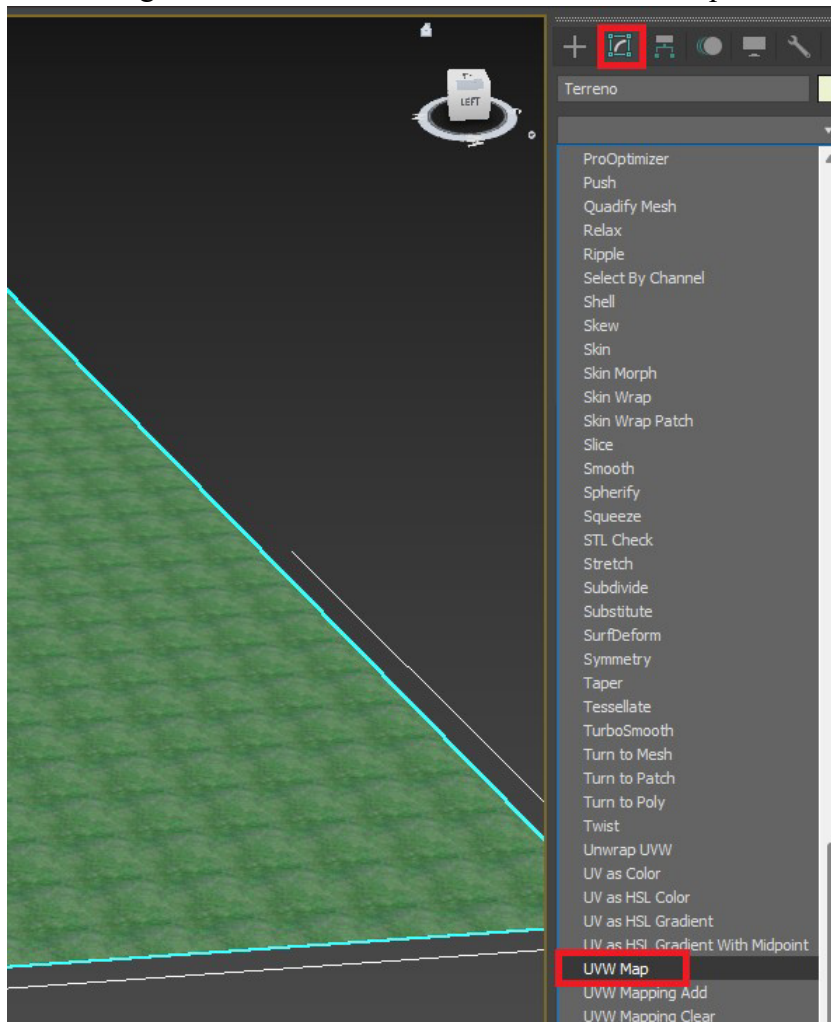


Fonte: Elaborada pelo autor.

3.1.1 Mapeamento de textura - Tile

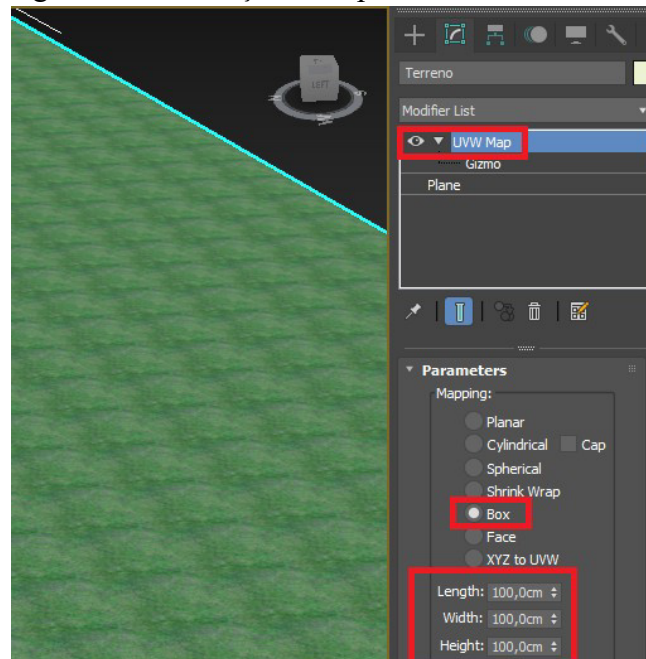
A modulação da textura do terreno foi desenvolvida em um metro por um metro, pois irá facilitar o ajuste posteriormente do Tile dessa textura dentro do Unreal. Assim, não precisará ficar exportando e importando a mesma mesh para conseguir ajustar o tamanho dessa textura de forma correta. Para usarmos esse tipo de formato foi necessário utilizar um modificador chamado de UVWmap (Figura 9), no qual nos dá por padrão o mapeamento de forma plana, porém foi modificado para o parâmetro Box para ser realizado os ajustes necessários na textura (Figura 10).

Figura 9 – Modelo sendo editado no UVW Map.



Fonte: Elaborada pelo autor.

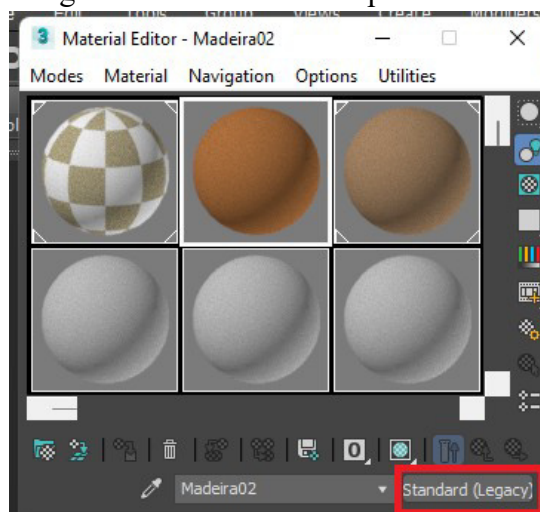
Figura 10 – Alterações nos parâmetros do material.



Fonte: Elaborada pelo autor.

Um detalhe importante para realizarmos uma boa exportação do projeto para Unreal é que os materiais da cena devem estar como tipo *standard* (Figura 11). No mundo real, a aparência de uma superfície depende de como ela reflete a luz, assim sendo, no 3ds Max, o formato *standard* simula as propriedades refletivas de uma superfície. Para que os materiais possam ter maior realismo na cena é necessário escolhermos suas cores e outras propriedades para que possa se assimilar como objetos do mundo real, outro fator importante é colocar os objetos em diferentes condições de iluminação.

Figura 11 – Material em tipo Standard.



Fonte: Elaborada pelo autor.

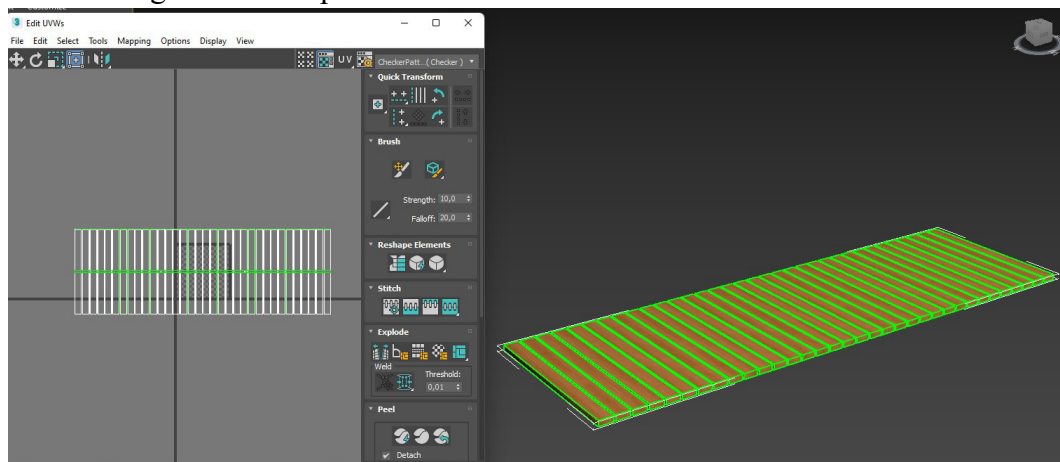
3.1.2 Mapeamento UV

O mapeamento UV é o processo em que a malha de um objeto é desenvolvida ou desdobrada e projetados sobre uma imagem 2D, em que cada área dessa imagem corresponde ao grupo de polígonos que será sobreposto (SORKINE; COHEN-OR, 2001). O processo de mapeamento UV em seu formato mais simples necessita de três etapas: desdobramento da malha, criação da textura e aplicação da textura. Assim sendo, os mapas gerados (também chamados de texturas) vão se ajustar na mesma posição dos pixels em todas as imagens 2D.

No software 3ds max foi utilizado dois canais UVs para o mapeamento, pois posteriormente quando este projeto for exportado para o software Unreal, o primeiro canal será responsável pela textura e o segundo para as informações de luz para utilizarmos no processo de *Bake*.

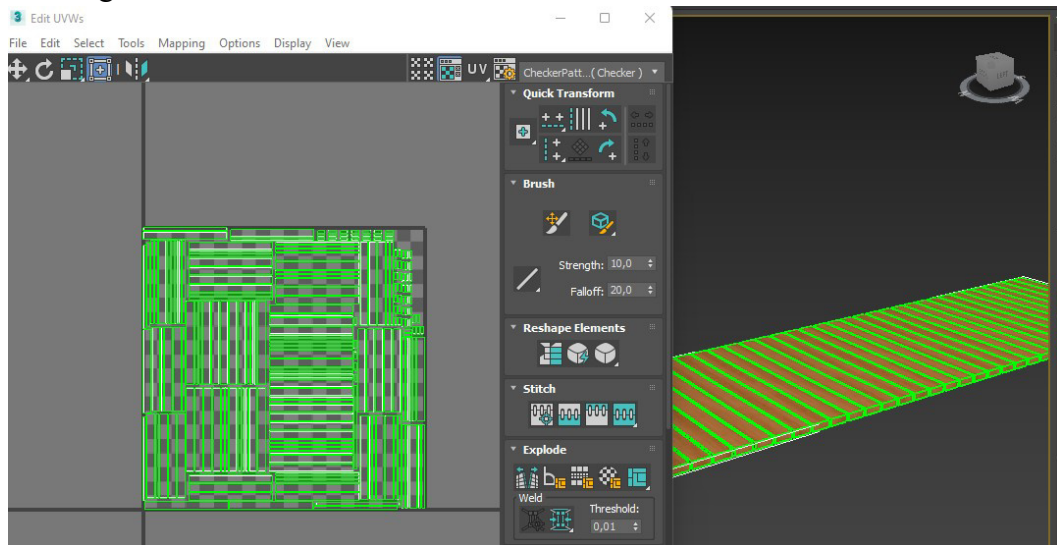
A Figura 12 mostra o tile da textura da madeira do deck no qual está sendo feita no canal 1 e a Figura 13 mostra a abertura de malha planificada do canal 2 disposta em uma plano bidimensional, no qual, a Unreal irá receber as informações de luz e sombra. Foi percebido que dentro da Unreal os canais são diferentes do que no 3ds max sendo eles 0 e 1.

Figura 12 – Mapeamento UV do modelo tridimensional do deck.



Fonte: Elaborada pelo autor.

Figura 13 – Desdobramento da malha do modelo tridimensional do deck.



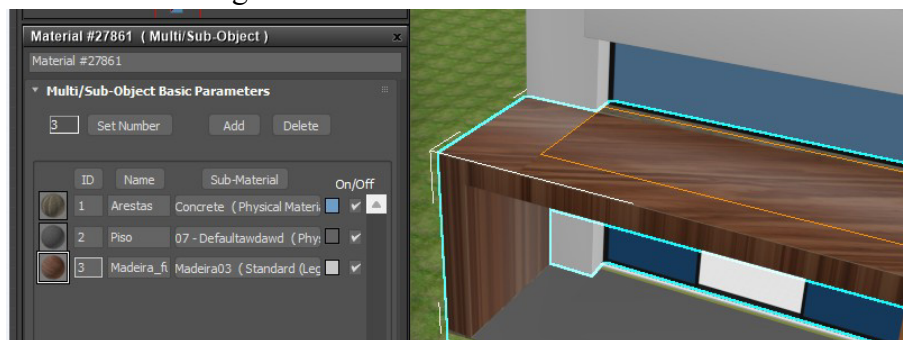
Fonte: Elaborada pelo autor.

Com o mapeamento da malha, a inserção de alguns materiais e texturas com tecnologia Physically Based Rendering (PBR) proporciona uma qualidade realística dos materiais baseados em física, evidenciando se ele é metálico ou não, se tem a superfície rugosa ou lisa, se reflete muita ou pouca luz, além de definir a sua cor base RGB (MCDERMOTT, 2018).

3.1.3 *Uso de ID para organizar os materiais*

A organização dos números de identificação das texturas que estão presentes nas meshes são chamadas de ID, é através delas que conseguimos identificar, na visualização do objeto, vários materiais presente no mesmo. A vantagem de utilizarmos essa forma de organização dos materiais ajudará na manipulação das texturas na Unreal para ser inseridos de forma individual de acordo com cada IDs criado no 3ds Max. Na (Figura 14) conseguimos entender melhor como se deu esse formato de organização.

Figura 14 – IDs do material de madeira.



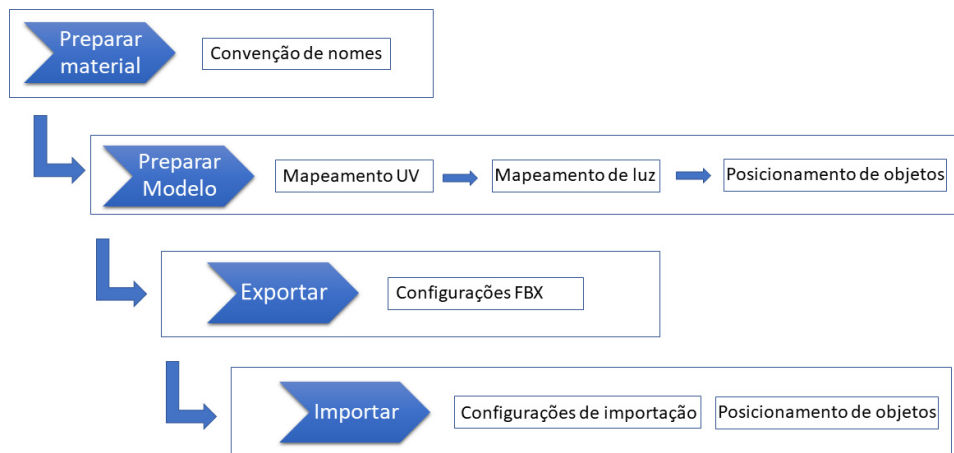
Fonte: Elaborada pelo autor.

Para obter esse resultado de vários materiais aplicados em um mesmo objeto, foi feito primeiramente a criação de uma estrutura no fundo feita de madeira e depois a transformação dessa estrutura em polígono e após essa etapa foi selecionado cada face para ser aplicado a textura, foi escolhido três tipos de materiais, nas faces o material aplicado foi a madeira, entre as divisórias das faces foi inserido o material chamado arestas e por último o material piso.

3.2 Workflow do FBX para Unreal

Neste projeto, como apresentado na Figura 15, foi adotado o formato de fluxo de produção do arquivo em Autodesk Filmbox (FBX) para a exportação e importação do modelo da casa, devido esse tipo de arquivo possibilitar a transferência de objetos 3D, dados relativos à textura de materiais, geometria do modelo, iluminação e manter a fidelidade e funcionalidade do arquivo original. Para fins de gerenciamento de assets, é uma boa prática nomear as static meshes com o prefixo de identificação, como “SM_”, isso irá garantir que o arquivo de malha estática tenha um identificador exclusivo.

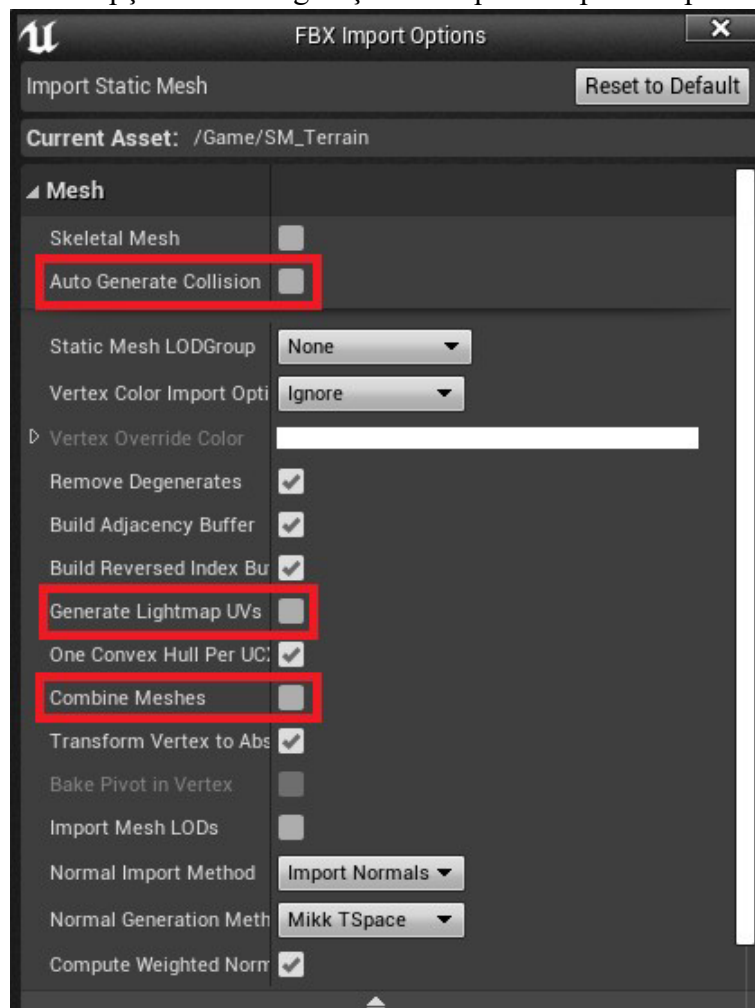
Figura 15 – Workflow do FBX.



Fonte: Elaborada pelo autor.

Após feito o procedimento de exportação do arquivo em FBX, o próximo passo é desmarcar algumas opções pré selecionadas de importação do FBX, como mostra na Figura 16, para receber esse arquivo dentro do Unreal, desmarcando facilitará posteriormente o processo manual de configuração dessas opções.

Figura 16 – Opções de configuração do arquivo importado pelo Unreal.



Fonte: Elaborada pelo autor.

3.3 Visualização do projeto após a importação para o Unreal Engine

Essa etapa consiste em organizar separadamente toda estrutura que estava separada por layer no 3ds max para dentro do content (diretório principal) da Unreal. Desse modo, todo o projeto ficará organizado para que os objetos, materiais e texturas sejam identificados de forma mais fácil. Após feito a inserção de todos os assets (nome dado ao conteúdo importado) dentro da interface Unreal) para suas respectivas pastas, foi realizado a seleção dos assets para viewport

(janela de visualização), ver Figura 17.

Figura 17 – Projeto da casa importado para o Unreal.

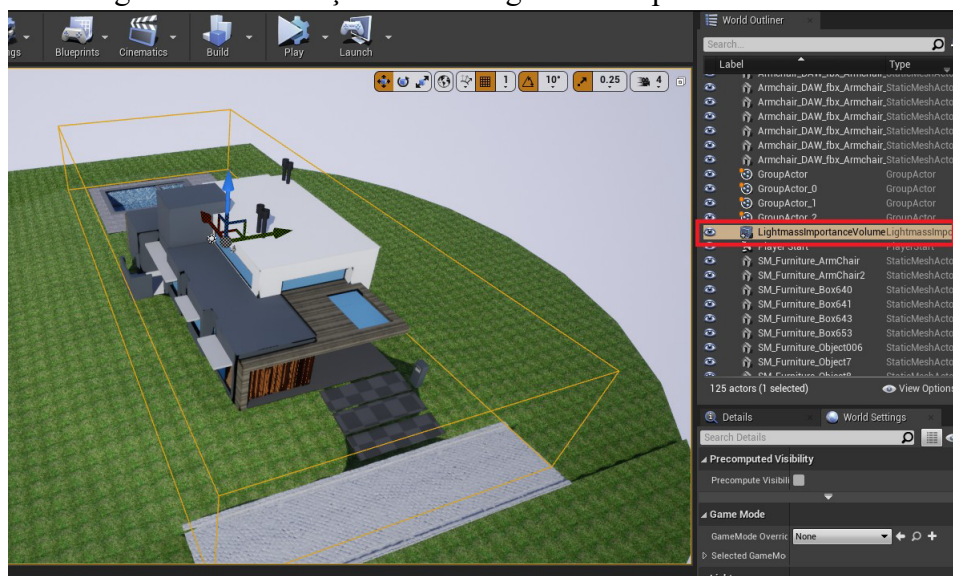


Fonte: Elaborada pelo autor.

3.3.1 Utilização do Lightmass

O Unreal Engine utiliza um sistema de iluminação global GI para calcular a iluminação indireta. Foi utilizado o ator chamado de “Lightmass importance volume” para delimitar toda a área da casa com o intuito de aumentar os cálculos mais precisos de luz, assim sendo, o tempo de cálculo será otimizado obtendo uma qualidade melhor na iluminação da área escolhida, como apresenta a Figura 18.

Figura 18 – Utilização do ator Lightmass importance volume.



Fonte: Elaborada pelo autor.

3.3.2 Definindo os níveis de qualidade no mapa do projeto

Os arquivos gerados do mapa contém todas as informações sobre o nível. Todo objeto do nível é referenciado como ator. Por exemplo, uma *static mesh* (malha estática) é inserida no nível, as informações adicionais são alocadas sobre o objeto, como propriedades de colisão, rotação, localização, etc. Neste projeto será gerado mapas de *levels* para poder fazer as comparações em critério de qualidade de renderização.

3.3.3 Build do modelo

Após ajustarmos o lightmass, foi feito os cálculos de luz em 3 níveis de qualidade que é oferecida pelo motor de jogo, a qualidade *preview*, na Figura 19, *medium*, *high* na Figura 20. Foi realizado duas *Build*(renderizações) para identificar de forma mais notória as mudanças após os cálculos de luz. Foi possível notar que foi resolvido pequenos problemas nas texturas dos objetos, porém ainda não ficou tão expressivo, tornando insuficiente para uma qualidade melhor, como apresentado na Figura 21.

Figura 19 – *Build* em qualidade preview.



Fonte: Elaborada pelo autor.

Figura 20 – *Build* em qualidade hight.



Fonte: Elaborada pelo autor.

Figura 21 – Defeitos presentes na textura após o *Build*

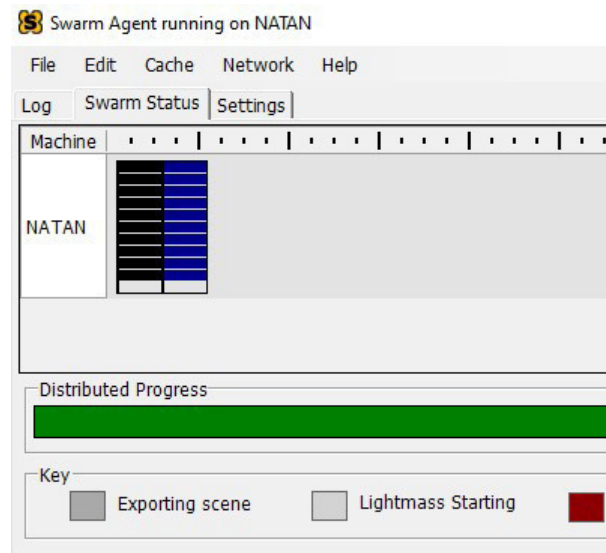


Fonte: Elaborada pelo autor.

Assim sendo, foi explorado outros métodos para melhorar a qualidade da cena, primeiramente foi ativado as configurações mundiais de Build , em “Settings” e depois em “World settings”. A partir disso, é percebido que o tempo para calcular o níveis de luz e rebatimento da cena é proporcional, ou seja, quando são aumentados os níveis o tempo de Build também é maior. A Unreal disponibiliza o progresso em tempo real e o tempo final de Build , através de

um programa chamado de “Swarm Agent”, que é aberto automaticamente a cada Build realizada, essa aplicação também é utilizada para configurar um sistema de gerenciamento de tarefas, no qual utiliza outras máquinas para ajudar no processo de renderização, mas no nosso caso não foi preciso essa configuração, ver Figura 22.2

Figura 22 – Tempo final de *Build* realizado pelo Swarm Agent.



Fonte: Elaborada pelo autor.

Na etapa realizada anteriormente foi capturado esse processo. Após feito os processos de Build o tempo gasto para realizar o primeiro cálculo foi de quinze segundos e após os ajustes de luz foi para trinta e quatro segundos. Então, pode ser percebido que a diferença foi bastante considerável, visto que a cena ainda possui poucos objetos. Quando foi feito as seguintes alterações o tempo final passou a ser de um minuto , ver Figura 23 e Figura 24.

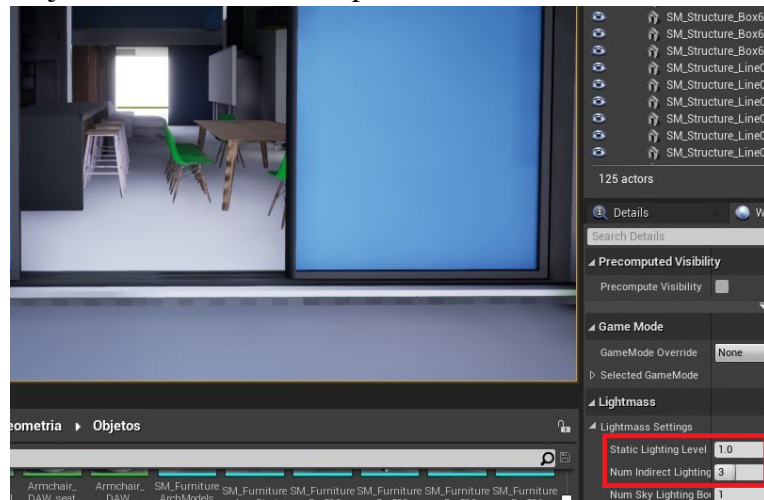
3.4 Volume de pós processamento

Foi percebido que quando se entrava dentro da casa, a luz começa a ficar com maior intensidade e quando é retornado para o lado de fora o sombreamento fica mais evidente, isso ocorre devido a física dentro do Unreal ser semelhante ao que acontece no mundo real, a pupila do olho humano dilata quando está dentro de um ambiente mais escuro, consequentemente a luz que é rebatida se torna mais intensa, quando é retornado para o lado de fora desse ambiente a vista começa a escurecer devido ao nosso olho contrair por conta da luminosidade. ³

² [https://docs.unrealengine.com/4.27/en-US/RenderingAndGraphics/Lightmass/UnrealSwarmOverview/#:~:text=static%20global,What%20is%20Unreal%20Swarm%3F,its%20assigned%20job\(s\)](https://docs.unrealengine.com/4.27/en-US/RenderingAndGraphics/Lightmass/UnrealSwarmOverview/#:~:text=static%20global,What%20is%20Unreal%20Swarm%3F,its%20assigned%20job(s))

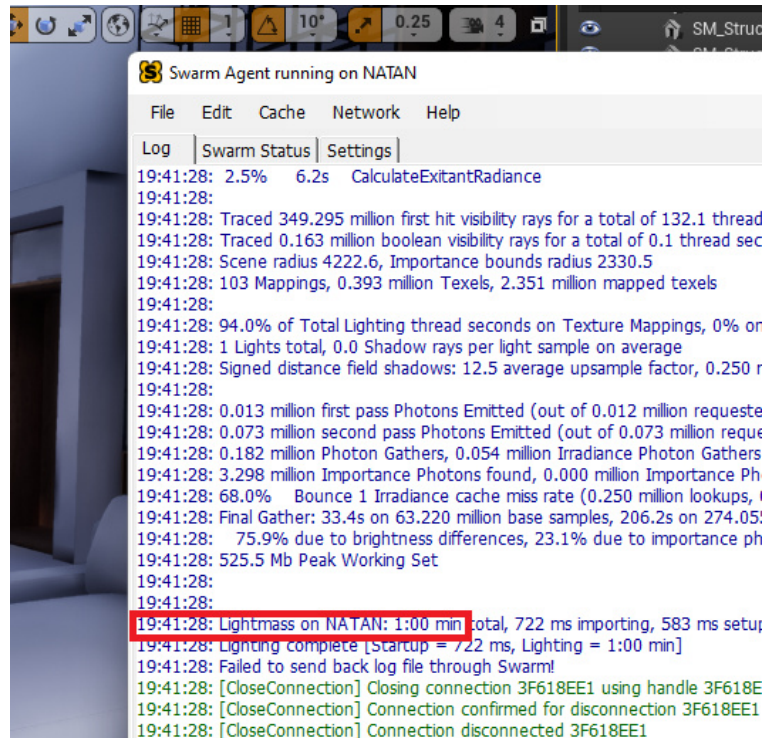
³ <https://vivaofthalmologia.com.br/pupila-dilatada-por-que-acontece-e-quando-se-preocupar/#:~:text=As%20pupilas%20se%20dilatam%20como,um%20ambiente%20com%20pouca%20ilumina%C3%A7%C3>

Figura 23 – Ajustes nos níveis de luz para melhorar a luminosidade do ambiente.



Fonte: Elaborada pelo autor.

Figura 24 – Tempo final de *Build* realizado pelo Swarm Agent.



Fonte: Elaborada pelo autor.

3.4.1 Luz Estática

A luz estática possui informações aprimoradas da luz direta, dos rebatimentos, sombras, etc. no qual armazena todas essas informações globais, além disso não há perda de desempenho, pois, quando é feito o *Build*, todas essas informações de luz são armazenadas durante esse processo, em contra partida, depois do *Build* não será possível alterar os ajustes de luz como sua intensidade, rotação, a sua cor e outras funcionalidades, para que seja feito é

necessário refazer os cálculos de luz.

3.4.2 *Luz Estacionária*

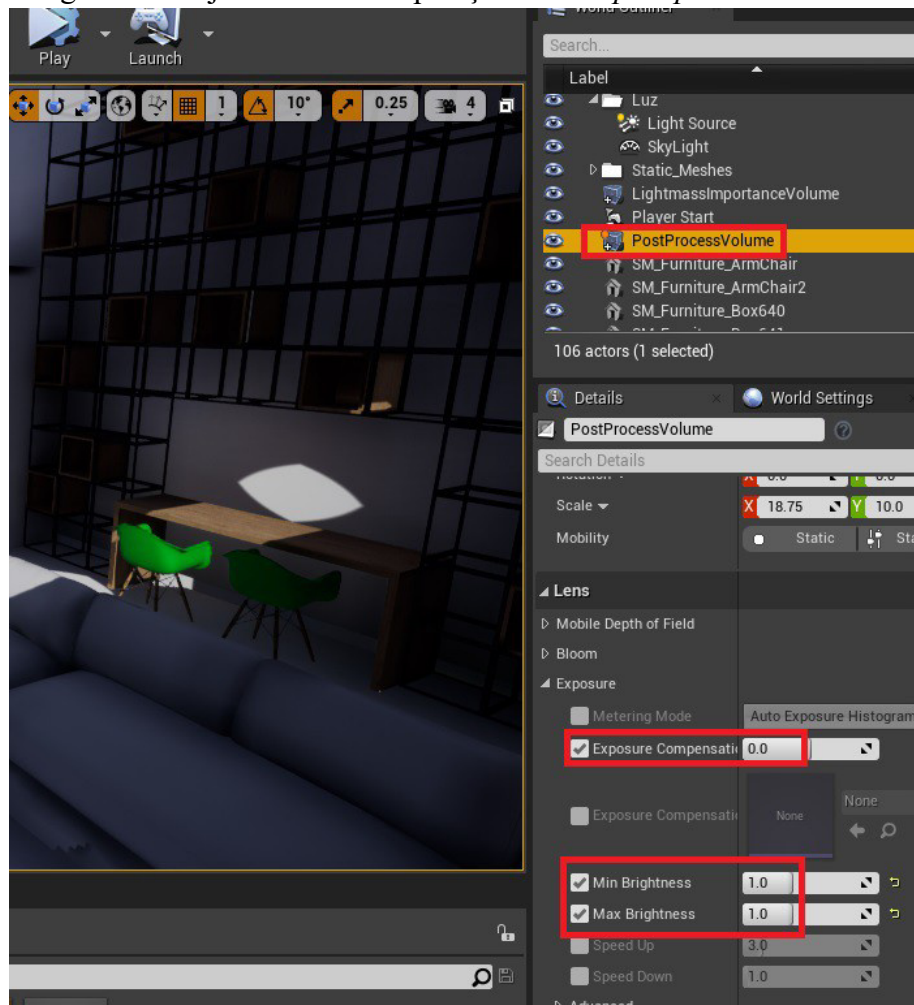
É o meio termo entre a luz estática e a móvel, no quais as informações de luz, rebatimentos, GI, sombra de contato, ajustes de cor e intensidade, mas sem movê-la de posição e terá uma influência mediana de desempenho, pois algumas características de cor será trabalhada em tempo real.

3.4.3 *Luz Móvel*

A luz móvel na Unreal é recomendada quando se trabalha com uma luz totalmente dinâmica devido a grande vantagem de trabalhar toda a cena em tempo real, com ajustes de intensidade, rotação, rebatimentos, sombras da luz direta entre outros. No entanto, há uma grande perda de desempenho pois toda a informação está sendo feita em tempo real e não há nenhuma informação de iluminação global, rebatimento, sombras e outras variáveis.

Desse modo, para o projeto da casa foi escolhido o tipo estático, pois não será necessário alterar o posicionamento do sol, nem a intensidade e a cor. Por ser um projeto pesado, devido a possuir muitos elementos em cena, o objetivo dessa escolha foi para armazenar as informações durante o *Build*. Para conseguir simular os recursos disponíveis na utilização do qual a luz móvel possui, foi necessário utilizar técnicas como Ambient Occlusion (AO) (para escurecer os cantos) e *mesh distance field*. Para ter um teste mais preciso de luz, foi utilizado o ator (objeto que pode ser colocado dentro janela de visualização) chamado de *post process volume* para controlar melhor essa autoexposição, em *Details* foi alterado os valores para zero da exposição e o de mínimo e máximo para 1, ver Figura 25.

Figura 25 – Ajustes da autoexposição do ator *post process volume*.



Fonte: Elaborada pelo autor.

3.4.4 Ajustes de luz indireta e sombra

Após os ajustes de exposição, foi trabalhado a luz indireta tendo sido feito ajustes (Figura 26) utilizando o ator *Sky Light* para ter a experiência mais próxima da realidade possível. Assim, foi percebido que, quando a luz de fora entra dentro da casa, acaba sendo rebatida nos objetos e paredes da cena, aumentando a luminosidade dela em seu interior. Ademais, também foi utilizado mapa HDRI (*i.e High Dynaminc Range Image*) que são imagens em formato de 360°, no qual possui informações de luminosidade, com a capacidade de simular fontes de luz diversas em um ambiente 3D, ver Figura 26 (PIERSON *et al.*, 2017).

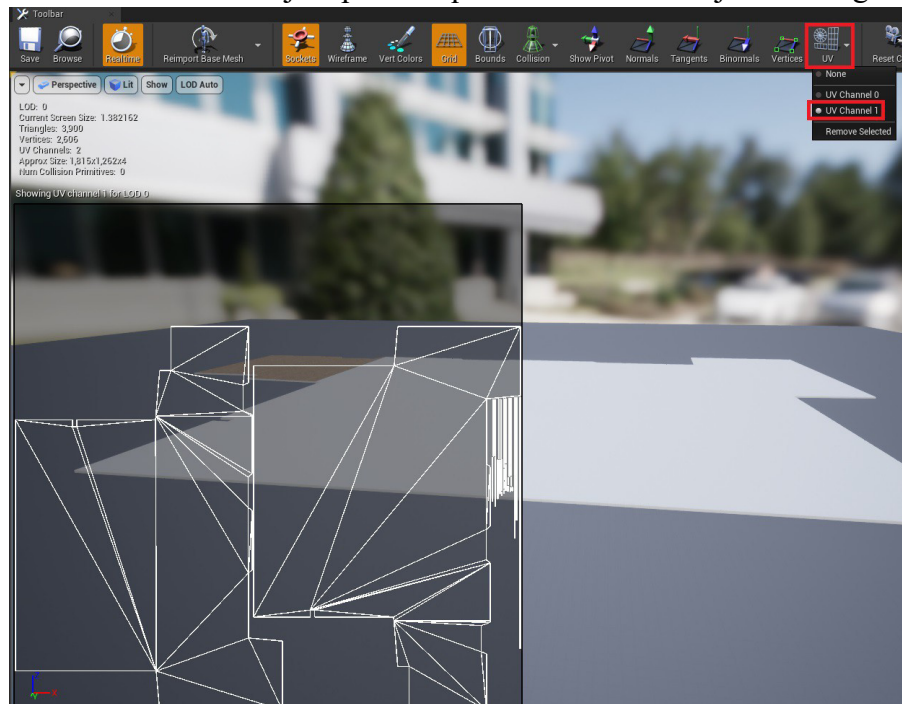
As sombras no qual foram feitos os ajustes de luz ficaram poucas definidas, sendo assim, para melhorar a qualidade da sombra da luz indireta foi aumentada a resolução do *light map* no qual será armazenado no canal UV 1 essas informações, como mostra a Figura 27. Nas opções de *stactic mesh* do piso foi feito os ajustes para para aumentar a resolução.

Figura 26 – Ajustes da luz indireta utilizando o ator Sky Light .



Fonte: Elaborada pelo autor.

Figura 27 – Canal 1 do objeto piso no qual foi recebido os ajustes de lightmap. .

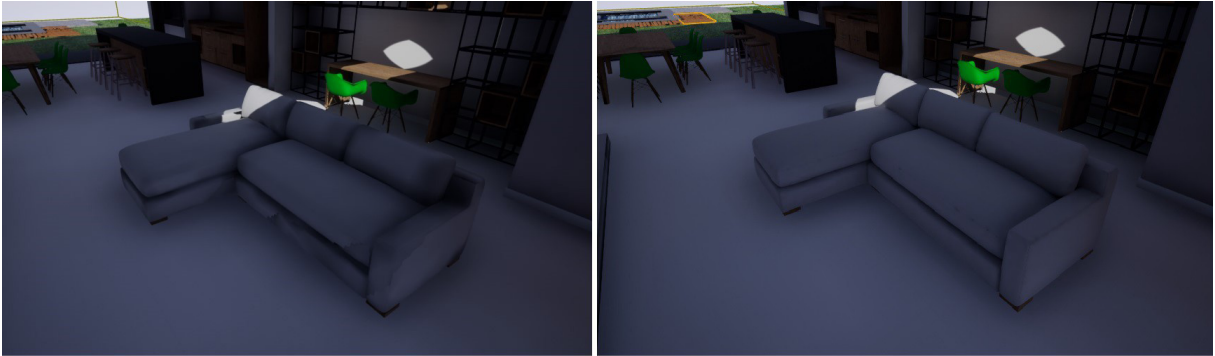


Fonte: Elaborada pelo autor.

Após feitos os ajustes, foi acompanhado o tempo de *Build*. Assim, notou-se que o tempo de *Build* foi prolongado devido a esses ajustes realizados. Como resultado das aplicações de resolução feitas, foram percebidas sombras mais definidas na cena, ver Figura 28. As resoluções mais altas impactam no espaço em disco adicional para os dados de textura no qual, conseqüentemente, demorou mais para a iluminação ser construída na cena. ⁴

⁴ <https://docs.unrealengine.com/5.0/en-US/static-light-mobility-in-unreal-engine/>

Figura 28 – Ajustes feitos no objeto piso para melhorar a resolução do lightmap.



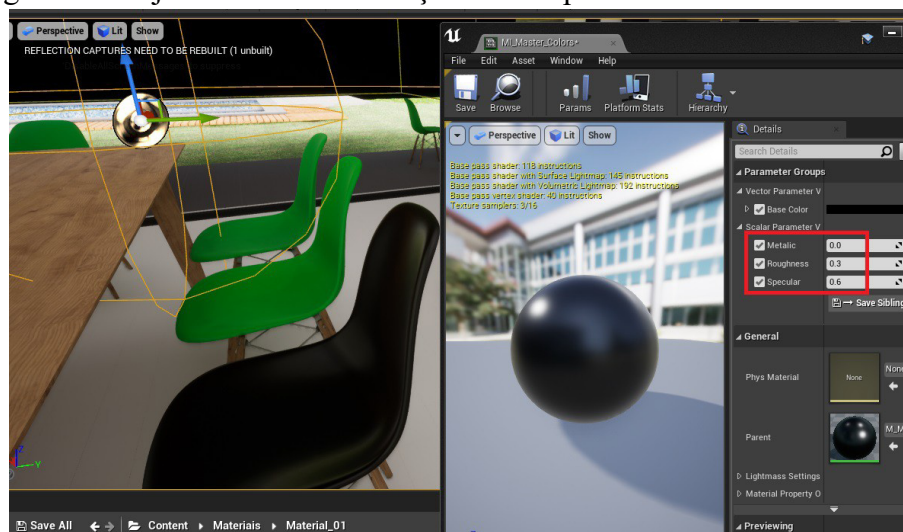
Fonte: Elaborada pelo autor.

3.4.5 Ajustes de Materiais

Foram criados os materiais para deixar a cena mais realista. No Unreal foi criado um material Mastere depois foram feitas instâncias desse material para ser possível trabalhar de forma mais leve e por convenção de nome, foi utilizado os prefixos “M_” para o “Master” e “MI_” para suas instâncias. Para a cadeira foi feito ajustes no material no qual possui detalhes reflexivos, ver Figura 29. Assim sendo, foi realizado ajustes nos parâmetros de *roughness* (nível de rugosidade), *specular* (nível de reflexo) e *metallic* (nível metálico) de ambos materiais para aproximar da imagem de referência, ver Figura 30.

Foi utilizado também um ator chamado *Sphere Reflection Capture*, esse recurso foi muito útil para oferecer reflexos mais realista para os materiais na cena. Foi aproveitado do material da cadeira pra fazer as instancias referentes aos outros objetos.

Figura 29 – Ajustes com renderização em tempo real do material da cadeira.



Fonte: Elaborada pelo autor.

Figura 30 – Imagem de referência.



Fonte: Marcelo Donadussi.

3.4.6 Renderização final do projeto

Foram feitos os ajustes necessários da aplicação dos materiais em seus respectivos objetos da cena, aplicação das texturas, ajustes de iluminação tanto interna quanto externa, foi também realizado a renderização em qualidade alta de todo o projeto, no qual resultou em um tempo total de 41 minutos, foi realizado a captura das imagens por câmeras, uma posicionada para fachada principal ver Figura 31 e sua referência, ver figura 32.

A Figura 33 possui uma vista da fachada posterior, e sua referência, ver Figura 34. A Figura 35 tem vista do interior da casa e sua referência, ver Figura 36. A figura 37 possui uma vista em diagonal da cozinha.

Figura 31 – Fachada principal da casa feita no Unreal Engine.



Fonte: Elaborada pelo autor.

Figura 32 – Vista da fachada posterior feita no Unreal Engine.



Fonte: Elaborada pelo autor.

Figura 33 – Vista do interior da casa feita no Unreal Engine.



Fonte: Elaborada pelo autor.

Figura 34 – Vista em diagonal da cozinha.



Fonte: Elaborada pelo autor.

3.5 A renderização em tempo real como solução tendo como finalidade uma representação fotorrealista

O ponto crucial da renderização em tempo real é que os usuários podem interagir com a renderização ao mesmo tempo em que ela é desenvolvida, dando ao usuário uma conexão com o espaço tridimensional (AKENINE-MÖLLER *et al.*, 2008). A partir dessa premissa, levando-se

em conta o ofício interativo e mutável de fazer uma obra, seja de arquitetura, paisagismos, construção civil, ou outra correlata, analisar um instrumento que permita modificações paulatinas ao processo de criação, ou seja, sem que haja demora, traz à tona uma alternativa moderna e eficiente para a produção arquitetônica.

Como exemplo, o ray tracing, uma das técnicas aqui abordadas, era usado apenas para a renderização offline devido ao seu custo computacional gerando demora no processo, nesses casos, focava-se na qualidade do resultado. Com o passar dos anos, com o desenvolvimento das CPU's e das GPU's, o ray tracing começou a ser utilizado em tempo real, ganhando amplo espaço no mercado gráfico (MELO, 2012).

A partir disso, depreende-se que os avanços ocorridos nos últimos tempos na computação gráfica possibilitaram a disseminação desses recursos de renderização. Atualmente, o uso de tecnologias de *game* possibilita a criação de modelos interativos rapidamente dentro de um determinado contexto, podendo ser um jogo, um protótipo automobilístico, um longa metragem ou até uma obra arquitetônica.

Os principais ganhos da renderização em tempo real é a possibilidade de mais interação com o usuário, a maior produtividade, a economia de tempo, já que as renderizações são instantâneas ou em tempo muito reduzido, *feedbacks* mais rápidos evitando alterações “em cima da hora” e utilização em várias plataformas.

4 CONSIDERAÇÕES FINAIS

A partir do que fora exposto, constata-se que o uso de tecnologias de video *game* no campo da Arquitetura, especialmente a renderização em tempo real, possibilita a realização de projetos eficientes e de fácil demonstração ao público em geral. Como evidenciado nas imagens aqui apresentadas, a visualização de imagens 3D confere realidade ao objeto visualizado, possibilitando uma melhor compreensão do conteúdo do projeto que será apresentado ao cliente.

Nesse contexto, a renderização tridimensional (3D) se mostra como um instrumento de design substancial para o diálogo entre arquitetos e clientes, possibilitando uma melhor interpretação visual quando se analisa desenhos arquitetônicos mais complexos. Nos projetos mais recentes, como existem muitos softwares renderizadores no mercado, existe um processo progressivo de adesão dos arquitetos em relação as novas tecnologias aqui apresentadas, mas, de forma mais tímida e até bem rara, quanto ao uso das tecnologias usualmente utilizadas em *game*, ou seja, técnicas mais aprofundadas de compartilhamento da imagem 3D.

A partir dessa análise, verificou-se que a maioria desses programas não são gratuitos e possuem código fechado, fato que acaba onerando o processo de elaboração do projeto arquitetônico, sendo este um obstáculo ainda a ser superado. Ademais, é necessário que seja divulgada a possibilidade da implementação dessa tecnologia “emprestada” dos *game* como uma alternativa útil para o melhoramento da imagem, no que cerne ao detalhismo e a agilidade no processamento, uma vez que os processos de renderização ainda estão muito limitados a rasterização e ao ray tracing mais simples.

Apropriar-se de uma tecnologia antes restrita aos jogos e aos longa-metragens de animação dos grandes estúdios de cinema inaugura novas possibilidades de experiência do universo da arquitetura e urbanismo, que se mostram cada dia mais aplicado em realidades virtuais, visto que muitos desses jogos de video *game* se moldam na arquitetura física, pois objetivam levar seus usuários a terem experiências e percepções cada vez mais realistas, como mencionado anteriormente. Do mesmo modo em que os *game* pegam emprestado essa arquitetura física mencionada para empregar na sua interface, a Arquitetura pede licença para se valer da tecnologia de ponta presente nos *game* e tornar a experiência do cliente mais satisfatória.

Ou seja, quando um profissional da arquitetura se propõem a utilizar um instrumento utilizado no ambiente dos *game*, está caracterizada a intenção de otimizar a experiência do usuário, tal como ocorre com a adesão e fascínio construída entre o jogo e o jogador. Assim, ambas as áreas extraem a qualidade da outra com a finalidade de ganhar o cliente.

A partir deste impulso, qual seja, levar uma experiência de qualidade ao cliente, este relatório se presta a evidenciar a qualidade de iluminação e reflexos obtidos com a renderização utilizando o *ray tracing* para renderização em softwares que utilizam tecnologias de game como Unreal, Blender Eevee, dentre outros, softwares comumente utilizados nos jogos, sendo essa renderização em tempo real. Nesse caso, utilizou-se o Unreal Engine, valendo-se de todas as qualidades de assets e habilidades peculiares ao software, conferindo realidade, precisão, detalhismo e rapidez ao projeto residencial ora compartilhado.

Como exemplo dessa implementação, a junção da tecnologia de *game* e da arquitetura não é novidade no meio acadêmico, na Universidade de Vila Velha no Espírito Santo existe um projeto, desde o ano de 2015, chamado “Cidades Virtuais”, tendo sido criado para ser executado nas turmas do curso de graduação de Arquitetura e Urbanismo, auxiliando na disciplina de Fundamentos Sociais, Econômicos e Ambientais de Arquitetura e Urbanismo.

O projeto possibilita a utilização do conhecido jogo *SimCity* para o incentivo da conexão dos conhecimentos teórico-conceituais da disciplina mencionada acima com práticas em ambiente virtual, além das demais disciplinas vistas durante o curso, aplicando os conhecimentos teóricos em ambiente virtual.

Ao analisar o produto ora renderizado pode ser percebido que a técnica da renderização é um importante instrumento para o trabalho do profissional da Arquitetura, haja vista que visualizar uma área externa de diferentes ângulos, testar paleta de cores e combinar *assets* para formar um padrão pedido pelo cliente auxilia o profissional a tomar uma decisão com maior segurança ainda mais quando a renderização é realizada em tempo real, gerando um produto cada vez mais fiel ao que fora planejado.

Para trabalhos futuros, propõem-se uma investigação de como o uso do *ray tracing* e do *path tracing*, esse último ainda em aperfeiçoamento e com grande potencial no cenário da computação gráfica, especificamente as sub técnicas peculiares aos video *game* s, estão sendo operacionalizadas na Arquitetura com a finalidade de manter o status das técnicas tradicionais da arquitetura mas abrindo espaço ao novo. Aqui é vislumbrada a implementação na seara da Arquitetura do que está sendo proposto nesse relatório, a partir da percepção dessa ideia como algo útil, inovador e otimizador dos processos.

Objetivar-se-á também a implementação de outras técnicas de renderização, a fim de adicionar mais elementos, tais como texturas e materiais, percebendo se o renderizador utilizado repara as falhas percebidas pelos profissionais de arquitetura e urbanismo. Sendo assimilado e

divulgado como um procedimento viável também para usuários de outras plataformas.

Outro ponto importante a ser mencionado é o potencial enxergado na versão 5 do software Unreal Engine, no qual este apresenta um recurso chamado “Lumen” que faz a simulação da luz refletida pela cena em tempo real reagindo imediatamente às mudanças de cena e luz de forma dinâmica. Desse modo, percebemos que com o avanço das tecnologias de *game*, novas formas de aprimoramento de técnicas em renderização em tempo real continuam em evolução, contribuindo para que trabalhos futuros possam ser aperfeiçoados.

REFERÊNCIAS

- AKENINE-MO, T.; HAINES, E.; HOFFMAN, N. *et al.* Real-time rendering. AK Peters/CRC Press, 2018.
- AKENINE-MÖLLER, T.; HAINES, E.; HAINES, E.; HOFFMAN, N. Real-time rendering: An ak peters book. CRC Press (Boca Raton, Fla.), 2008.
- APPEL, A. Some techniques for shading machine renderings of solids. In: **Proceedings of the April 30–May 2, 1968, spring joint computer conference**. [S.l.: s.n.], 1968. p. 37–45.
- BARATTO, R. **A imagem fala ou, Por que precisamos ir além dos renders | Passei Direto**. 2016. <<https://www.passeidireto.com/arquivo/50207499/baratto-romullo-a-imagem-fala-ou-por-que-precisamos-ir-alem-dos-renders>>. (Accessed on 06/24/2022).
- BOAS, N. V. Além da imagem estática: A representação gráfica digital da experiência espacial na arquitetura. In: **Visión y Visualización. IX Congreso Iberoamericano de Gráfica Digital (Sigradi). Anais**. Lima: **Universidad Peruana de Ciências Aplicadas (UPC)**. [S.l.: s.n.], 2005.
- BRAIDA, F.; LIMA, F.; FONSECA, J.; MORAIS, V.; GASPAR, J. **101 conceitos de arquitetura e urbanismo na era digital**. [S.l.]: ProBooks, 2017.
- BROOKER, G.; STONE, S. O que é design de interiores. **São Paulo: Senac**, 2014.
- CHEN, L. Architectural visualization: An analysis from human visual cognition process. **Program in Digital Arts & Design, Faculty of Art & Design, Monash University**, 2004.
- CHING, F. D. **Representação gráfica em arquitetura**. [S.l.]: Bookman Editora, 2000.
- DINUR, E. **The Complete Guide to Photorealism: For Visual Effects, Visualization and Games**. [S.l.]: Routledge, 2021.
- FERWERDA, J. A. Three varieties of realism in computer graphics. In: SPIE. **Human vision and electronic imaging viii**. [S.l.], 2003. v. 5007, p. 290–297.
- GOMES, P.; PAMPLONA, V. M3ge: um motor de jogos 3d para dispositivos móveis com suporte a mobile 3d graphics api. In: **Proceedings of the Brazilian Symposium on Computer Games and Electronic Entertainment**. [S.l.: s.n.], 2005. p. 55–65.
- JOLY, M. **Introdução à análise da imagem**. [S.l.]: Papirus editora, 1996.
- LEWIS, M.; JACOBSON, J. Game engines. **Communications of the ACM**, v. 45, n. 1, p. 27, 2002.
- LOPES, J. M. B. Computação gráfica: Rasterização. **Universidade Técnica da Lisboa**, 2013.
- MCDERMOTT, W. **The PBR guide**. [S.l.]: Allegorithmic, 2018.
- MELO, D. Lemos de A. **Um pipeline para renderização fotorrealística de tempo real com ray tracing para a realidade aumentada**. Dissertação (Mestrado) — Universidade Federal de Pernambuco, 2012.

- MORAIS, V. Renderização digital. **BRAIDA, Frederico et al.(Org.)**, v. 101, 2016.
- NERY, A. S. **GRIDRT: UMA ARQUITETURA PARALELA PARA RAY TRACING UTILIZANDO VOLUMES UNIFORMES**. Tese (Doutorado) — Universidade Federal do Rio de Janeiro, 2010.
- NVIDIA. Nvidia developer. **NVIDIA RTX™ platform**, 2018. Disponível em: <<https://developer.nvidia.com/rtx>>. Acesso em: 16 jul.2022.
- OLIVEIRA, E. R. de. O uso de engines para o desenvolvimento de jogos eletrônicos. **UNIVERSIDADE ESTADUAL DO SUDOESTE DA BAHIA - UESB**, 2013.
- PEDDIE, J. The global market for ray tracing software. **The Global Market for Ray Tracing Software**, 2019.
- PIERSON, C.; WIENOLD, J.; JACOBS, A.; BODART, M. **Luminance maps from High Dynamic Range imaging: calibrations and adjustments for visual comfort assessment**. [S.l.], 2017.
- PINHAL, P. S. 11. reflexões entre o desenho, representação gráfica e projeto arquitetônico. **Revista Científica UMC**, v. 2, n. 1, 2017.
- RIGHETTO, A. V. D. O desenho de arquitetura e seu desenho no tempo. **Artigo–Programa de Pós-Graduação da Faculdade de Arquitetura e Urbanismo da Universidade de São Paulo, São Paulo, sem data**, 2005.
- SANTAELLA, L.; NÖTH, W. **Imagem: cognição, semiótica, mídia**. [S.l.]: Iluminuras, 2020.
- SANTOS, A. P. B. d. Por uma arquitetura virtual: uma crítica das tecnologias digitais'. **Revista AU, Arquitetura e Urbanismo, São Paulo**, n. 131, p. 57–60, 2005.
- SANTOS, H. D. d. *et al.* Adaptação da série "dias velho" para renderização na unreal engine 4. Florianópolis, SC, 2018.
- SILVA, F. W. S. V. da. Introdução ao ray tracing. 1994.
- SOARES, C. C. P. Uma abordagem histórica e científica das técnicas de representação gráfica. **UFRJ-Universidade Federal do Rio de Janeiro, Departamento de Técnicas de Representação**, 2007.
- SONTAG, S. Sobre la fotografía, trad. **Rubens Figueiredo–São Paulo: Companhia das Letras**, 2004.
- SORKINE, O.; COHEN-OR, D. Warped textures for uv mapping encoding. In: **Eurographics (Short Presentations)**. [S.l.: s.n.], 2001.
- STEINGRÄBER, R. S. **Protótipo de um Sistema de Customização 3D para Planejamento de Edificações**. Tese (Doutorado) — Universidade Regional de Blumenau, 2003.
- ZANCANELI, M. A.; CHAGAS, I.; BRAIDA, F.; FERREIRA, I. de M. Os softwares de realidade virtual para o projeto de arquitetura na era da quarta revolução industrial: uma análise comparativa. **SIMPÓSIO BRASILEIRO DE TECNOLOGIA DA INFORMAÇÃO E COMUNICAÇÃO NA CONSTRUÇÃO**, v. 2, p. 1–6, 2019.
- ZEVI, B. **Saber ver a arquitetura**. [S.l.]: Martins Fontes São Paulo, 1996.